

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

*Кафедра автоматизованих систем обробки інформації та управління*

УДК 004.42

«До захисту допущено»

**В.о. завідувача кафедри**

О.А.Павлов  
(ініціали, прізвище)

“ ” \_\_\_\_\_ 2019 р.

**Дипломний проект**  
**на здобуття ступеня бакалавра**

з напрямку підготовки 6.050101 «Комп'ютерні науки»

на тему: «Інформаційна система моніторингу та екологічної  
оцінки водних ресурсів України»

**Виконала:**

студентка 4 курсу, групи ІС-51

Казмірчук Аліна Василівна

(прізвище, ім'я, по батькові)

(підпис)

**Керівник**

доц., к.т.н. Сперкач М.О.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

**Консультант з  
графічної  
документації**

доц., к.т.н., доц. Тєлишева Т.О.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

**Рецензент**

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цьому дипломному проекті  
немає запозичень з праць інших авторів без  
відповідних посилань.

Студент (-ка) \_\_\_\_\_  
(підпис)

Київ – 2019 року

*5. Технологічний розділ: керівництво користувача, методика випробувань програмного продукту*

## 5. Перелік графічного матеріалу

1. Схема структурна варіантів використання

2. Схема структурна послідовності

3. Схема структурна діяльності

4. Схема структурна компонентів

5. Схема бази даних

6. Схема структурна класів програмного забезпечення

7. Рішення з математичного забезпечення

8. Креслення вигляду екранних форм

## 6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання «15» лютого 2019 року

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1.	Вивчення рекомендованої літератури	17.04.2019	
2.	Аналіз існуючих методів розв'язання задачі	18.04.2019	
3.	Постановка та формалізація задачі	19.04.2019	
4.	Розробка інформаційного забезпечення	21.04.2019	
5.	Алгоритмізація задачі	25.04.2019	
6.	Обґрунтування використовуваних технічних засобів	26.04.2019	
7.	Розробка програмного забезпечення	18.05.2019	
8.	Налагодження програми	19.05.2019	
9.	Виконання графічних документів	23.05.2019	
10.	Оформлення пояснювальної записки	29.05.2019	
11.	Подання ДП на попередній захист	30.05.2019	
12.	Подання ДП на основний захист	03.06.2019	
13.	Подання ДП рецензенту	05.06.2019	

Студент

\_\_\_\_\_ А.В. Казмірчук  
(підпис)

Керівник проекту

\_\_\_\_\_ М.О. Сперкач  
(підпис)

[illegible]

## **Пояснювальна записка до дипломного проекту**

на тему: Інформаційна система моніторингу та екологічної оцінки  
водних ресурсів України

---

Київ – 2019 року

## АНОТАЦІЯ

**Структура та обсяг роботи.** Пояснювальна записка дипломного проекту складається з п'яти розділів, містить 54 рисунки, 36 таблиць, 1 додаток, 21 джерело.

Дипломний проект присвячений розробці інформаційної системи для підтримки процесу проведення моніторингу та екологічної оцінки водних ресурсів України.

Розділ загальних положень присвячений опису предметного середовища, дослідженню процесів діяльності. Описано функціональну модель, наведено огляд наявних аналогів, визначено призначення розробки, сформульовано цілі та задачі розробки.

У розділі інформаційного забезпечення описано вхідні та вихідні дані, наведено перелік доступних звітів та макети деяких з них, наведено схему структурну баз даних та опис її таблиць.

Розділ математичного забезпечення присвячений дослідженню ефективності методів прогнозування стану водних ресурсів України.

Програмне забезпечення описує обрані засоби для розробки. Наведено архітектуру програмного забезпечення, схему класів, послідовності та компонентів. Описано специфікацію основних методів продукту.

У технологічному розділі наведено керівництво користувача з покроковою інструкцією та екранними формами. Проведено випробовування програмного продукту.

ІНФОРМАЦІЙНА СИСТЕМА, МОНІТОРИНГ, НОРМАТИВИ, ДЕРЖВОДАГЕНСТВО, ПОСТ СПОСТЕРЕЖЕННЯ, ПРОГНОЗУВАННЯ, ЕКСТРАПОЛЯЦІЯ, НЕЙРОННІ МЕРЕЖІ, ЧАСОВИЙ РЯД

					ДП ІС-5109.1181-с.ПЗ				
		Прізвище	Підпис	Дата					
Розроб.		Казмірчук А.В.			Інформаційна система моніторингу та екологічної оцінки водних ресурсів України	Літ.	Лист	Листів	
Перевірив.		Сперкач М.О.					2	133	
						КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51			
Н. кон.		Телишева Т.О.							
Затв.		Павлов О.А.							

## ABSTRACT

**Structure and scope of work.** Explanatory note of the diploma project consists of five sections, contains 54 drawings, 36 tables, 1 supplement, 21 source.

The diploma project focuses on the development of an information system to support the process of monitoring and environmental assessment of water resources in Ukraine.

The section of the general provisions focuses on the description of the subject environment, the study of the processes of the State Water Management and water management organizations. The functional model is described, the review of existing analogues is given, the designation of the development is defined, the goals and tasks of the development are formulated.

The section of the information support describes the input and output data, reports, provides a schema of structure of the databases and a description of its tables.

The section of mathematical support is dedicated to the study of the effectiveness of forecasting methods for water resources in Ukraine.

The software section describes the selected development tools. The software architecture is presented using a structural class diagram, sequence, and components. The specification of the main software methods is described.

The technology section provides user guide with step-by-step instructions and on-screen forms. Software product testing has been performed.

INFORMATION SYSTEM, MONITORING, NORMATIVES, WATER AGENCY, MONITORING POST, FORECASTING, EXTRACTION, NEURAL NETWORKS, TIME RANGE

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

## ЗМІСТ

ВСТУП.....	5
1 ЗАГАЛЬНІ ПОЛОЖЕННЯ .....	6
1.1 ОПИС ПРЕДМЕТНОГО СЕРЕДОВИЩА .....	6
1.1.1 Опис процесу діяльності.....	7
1.1.2 Опис функціональної моделі.....	8
1.2 ОГЛЯД НАЯВНИХ АНАЛОГІВ .....	12
1.3 ПОСТАНОВКА ЗАДАЧІ .....	14
1.3.1 Призначення розробки.....	14
1.3.2 Цілі та задачі розробки .....	14
Висновок до розділу .....	16
2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ .....	17
2.1 ВХІДНІ ДАНІ .....	17
2.2 ВИХІДНІ ДАНІ.....	20
2.3 ОПИС СТРУКТУРИ БАЗИ ДАНИХ.....	26
Висновок до розділу .....	45
3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ.....	46
3.1 ЗМІСТОВНА ПОСТАНОВКА ЗАДАЧІ .....	46
3.2 МАТЕМАТИЧНА ПОСТАНОВКА ЗАДАЧІ .....	46
3.3 ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ’ЯЗАННЯ.....	46
3.4 ОПИС МЕТОДІВ РОЗВ’ЯЗАННЯ .....	47
3.4.1 Методи апроксимації.....	47
3.4.2 Нейронна мережа.....	49
3.4.3 Оцінка точності прогнозів .....	52
3.5 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ .....	52
Висновок до розділу .....	56
4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ .....	57
4.1 ЗАСОБИ РОЗРОБКИ.....	57
4.2 ВИМОГИ ДО ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ.....	59
4.2.1 Загальні вимоги .....	59
4.3 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	60



4.3.1	Діаграма класів .....	61
4.3.2	Діаграма послідовності .....	62
4.3.3	Діаграма компонентів .....	62
4.3.1	Специфікація функцій .....	62
4.4	ОПИС ЗВІТІВ .....	75
	Висновок до розділу .....	79
5	ТЕХНОЛОГІЧНИЙ РОЗДІЛ .....	80
5.1	КЕРІВНИЦТВО КОРИСТУВАЧА .....	80
5.2	ВИПРОБУВАННЯ ПРОГРАМНОГО ПРОДУКТУ .....	100
5.2.1	Мета випробувань .....	100
5.2.2	Загальні положення .....	100
5.2.3	Результати випробувань .....	101
	Висновок до розділу .....	102
	ЗАГАЛЬНІ ВИСНОВКИ .....	103
	ПЕРЕЛІК ПОСИЛАНЬ .....	105
	ДОДАТОК А .....	108

## ВСТУП

Проблема забруднення навколишнього середовища є сьогодні актуальною як ніколи, особливо це стосується водних ресурсів. Викиди промислових підприємств, техногенні аварії, каналізаційні стоки, купи сміття знищують ріки, озера, моря, перетворюючи воду в небезпечну для використання та непридатною для життя живих організмів.

Людство усвідомлює цю проблему і намагається вживати заходів для захисту та відновлення водних ресурсів. Прийняття рішень щодо розвитку і захисту водних об'єктів зазвичай базується на даних моніторингу, що дозволяють оцінити поточний стан водойм і зробити прогноз на майбутнє.

Також водогосподарські організації проводять контроль водогосподарської обстановки на водосховищах, водогосподарських системах для перевірки відповідності поточного стану нормативним вимогам та розрахунку відповідних параметрів.

Саме тому виникає необхідність у створенні системи для підтримки процесу моніторингу забрудненості водних ресурсів України, а також контролю водогосподарської обстановки на водосховищах, водогосподарських системах, контрольних пунктах. Це прискорить процес отримання актуальних даних моніторингу, спростить підтримку систематизації даних та підвищить ефективність роботи відповідних водогосподарських організацій.

Дипломний проект присвячений розробці для підтримки процесу проведення моніторингу та екологічної оцінки водних ресурсів України.

**Практичне значення одержаних результатів.** Досліджено методи проведення ефективної екологічної оцінки водних ресурсів України.

**Публікації.** Результати роботи були опубліковані у статті «Дослідження методів проведення ефективної екологічної оцінки водних ресурсів України» у науковому журналі «Innovative Solutions in Modern Science» [1].

					ДП ІС-5109.1181-с.ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

# 1 ЗАГАЛЬНІ ПОЛОЖЕННЯ

## 1.1 Опис предметного середовища

Державне агентство водних ресурсів України є центральним органом виконавчої влади, діяльність якого контролюється Кабінетом Міністрів України через Міністра екології та природних ресурсів. Держводагентство реалізує державну політику у сфері розвитку водного господарства, управління, використання та відтворення поверхневих водних ресурсів відповідно до законів України [2].

Держводагенству підпорядковані обласні управління водних ресурсів, 9 басейнових управлінь водних ресурсів (Західного Бугу, Тиси, Дунаю, Дністра-Прута, Південного Бугу, Десни, Росі, Дніпра, Сіверського Дінця), 5 управлінь каналів (Головного Каховського, Дніпро-Донбас, Дніпро-Інгулець, Північно-Кримський, каналів Інгулецької зрошувальної системи) та інші [3].

Держводагенство виступає центральним органом, що отримує загальні дані моніторингу по постах спостереження від усіх підлеглих водогосподарських організацій для подальшого аналізу і прийняття рішень.

Водогосподарські організації проводять моніторинг річок, водосховищ, каналів, зрошувальних систем і водойм у межах водогосподарських систем комплексного призначення, систем водопостачання, транскордонних водотоків та водойм у зонах впливу атомних електростанцій. Контроль якості води за фізичними та хімічними показниками здійснюється на 72 водосховищах, 164 річках, 14 зрошувальних системах, 1 лимані та 5 каналах комплексного призначення. Крім того, у рамках радіаційного моніторингу вод водогосподарськими організаціями здійснюється контроль вмісту радіонуклідів у поверхневих водах [4].

Догляд за поверхневими водами України та проведення заходів для екологічного оздоровлення є однією з найголовніших місій

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

Держводагентства. Особливо гостро це питання постає сьогодні, коли рівень забруднення навколишнього середовища сягає критичних показників.

Не дивно, що дані моніторингу екології викликають високий суспільний інтерес, адже майже 80% населення України отримує питну воду з поверхневих джерел, серед них майже 75% – із Дніпра. Саме тому людям важливо усвідомлювати, якої якості є поверхневі води поблизу їхнього дому, роботи чи місць відпочинку. Ці дані також дадуть можливість громадським активістам краще розуміти вплив промислових відходів на навколишнє середовище та контролювати дотримання законодавства в сфері екології.

Дані про якість води – одні з найбільш цінних, але водночас – і одні з найбільш закритих у світі. Відомо, що лише 15 країн у світі публікують ці дані у відкритому форматі [5].

Саме тому виникає необхідність у створенні єдиної системи моніторингу та екологічної оцінки водних ресурсів України, яка дозволяє зручно, у єдиному форматі зберігати дані моніторингу постів спостереження, що розташовані по всій території країни, відображає дані у зручному для аналізу вигляді з можливістю візуалізації динаміки змін показників та формування звітності. Це підвищить ефективність роботи екологів та відповідних служб, що є відповідальними за стан водойм України та надасть можливість суспільству отримувати актуальну інформацію про стан екології.

### 1.1.1 Опис процесу діяльності

Опишемо процес моніторингу водних ресурсів України, що здійснюють водогосподарські організації. В Україні діє багато водогосподарських організацій. Це, наприклад, Басейнове управління водних ресурсів річки Прип'ять, Деснянське басейнове управління водних ресурсів, Київський РУВР, Регіональний офіс водних ресурсів річки Рось, Регіональний офіс водних ресурсів у Волинській області, Управління каналу Дніпро-Донбас, Міжрегіональний офіс захисних масивів дніпровських

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

водосховищ, Басейнове управління водних ресурсів річок Причорномор'я та нижнього Дунаю. Для зручності опису розглянемо процес моніторингу на прикладі двох суб'єктів моніторингу, та Держводагенства. Нехай Басейнове управління водних ресурсів річок Причорномор'я та нижнього Дунаю - суб'єкт моніторингу 1, що здійснює моніторинг рівня забрудненості водних об'єктів, а Міжрегіональний офіс захисних масивів дніпровських водосховищ - суб'єкт моніторингу 2, що контролює водогосподарську обстановку. Всі дії відображає схема структурна діяльності, що наведена в частині графічного матеріалу.

Схема структурна діяльності, що наведена в частині графічного матеріалу відображає процес моніторингу та екологічної оцінки водних ресурсів України з використанням розробленої системи. Значення показників вимірюються як і раніше, проте тепер вже суб'єкти моніторингу вносять їх відразу до системи через мережу Інтернет і дані зберігаються в єдиній базі даних у визначеному форматі.

Схема структурна діяльності, що наведена в частині графічного матеріалу описує процес додавання даних моніторингу якості води по постах спостереження.

Схема структурна діяльності, що наведена в частині графічного матеріалу описує процес контролю водогосподарського стану.

### 1.1.2 Опис функціональної моделі

Опишемо функціональну модель системи за допомогою схеми структурної використання. Для цього визначимо дійових осіб (акторів) та дії, які кожен з акторів може виконувати в системі. В системі маємо 3 актори: Адміністратор, Співробітник, Користувач. Нижче наведений опис кожного з акторів.

**Адміністратор.** Співробітник організації, що наповнює довідники

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

системи, вносить значення нормативів, гранично допустиму концентрацію показників (ГДК) та керує користувачами.

**Співробітник.** Зареєстрований користувач системи, що може створювати, редагувати пости спостереження, вносити значення показників до таблиць спостережень, контролювати водогосподарську обстановку. Може переглядати карту перевищення ГДК показників якості води на постах спостереження, карту розміщень постів спостереження та формувати звітність за довільний період часу.

**Користувач.** Незареєстрований користувач, який може переглядати карту перевищення ГДК показників якості води на постах спостереження, карту розміщень постів спостереження та формувати звітність за довільний період часу без необхідності входу в систему.

Дії, які можуть виконувати актори системи, визначені на схемі структурній використання, що наведена в частині графічного матеріалу.

Розглянемо детальніше деякі варіанти використання:

– реєстрація пункту спостереження:

- 1) внесення даних - опис місця розташування, область, район річкового басейну, лінійна координата, назва головної лабораторії, назва водогосподарської організації, статус пункту спостереження та інші;
- 2) відображення пункту спостереження на карті – внесення географічних координат пункту спостереження;
- 3) визначення показників або групи показників для подальшого моніторингу;

– моніторинг пункту спостереження (ПС):

- 1) вибір ПС, що досліджується;
- 2) внесення вимірних значень показників якості води – створення нового запису в таблиці спостережень із

					ДП ІС-5109.1181-с.ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

зазначенням дати спостереження та заповнення рядка вимірними значеннями показників;

– контроль водогосподарської обстановки:

- 1) внесення даних по водогосподарським системам – внесення середньодобового забору;
- 2) внесення даних по гідрологічній ситуації на водосховищах - вибір водосховища, внесення фактичного рівня, фактичного припливу, фактичного скиду;
- 3) внесення даних по гідрологічній ситуації на контрольних пунктах – вибір річки-пункту, внесення фактичного рівня, фактичної витрати;
- 4) внесення даних гідрохімічної обстановки – вибір річки-пункту, вибір показника, внесення фактичного значення;

– заповнення ГДК показників:

- 1) вибір групи показників;
- 2) вибір показника;
- 3) внесення нормативних значень – мінімально або максимально гранично допустима концентрація;

– внесення нормативів:

- 1) внесення норм заборів води на водогосподарських системах – вибір каналу, зрошувальної системи, внесення декадного плану, внесення сезонів роботи із зазначенням сезону, дати початку і дати кінця сезону;
- 2) внесення координат кривих водосховища – вибір основного басейну, водосховища, внесення проектного рівня, проектного обсягу, внесення координат кривих водосховища із зазначенням відмітки, об'єму і вільного об'єму;
- 3) внесення нормативів рівнів та витрат води на контрольних пунктах - вибір річки-пункту, внесення норм витрат по

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

- місяцях, внесення норм рівнів по місяцях із зазначенням чисел початку і кінця декади, максимального і мінімального рівня;
- 4) внесення нормативів гіdroхімічних показників – вибір показника і внесення нормативного значення;
- перегляд карти перевищення ГДК показників на постах спостереження:
- 1) перегляд загальної карти перевищення ГДК показників, на якій кольоровими індикаторами позначені пункти спостереження, де колір індикатора відображає умови та якість води (рівень перевищення ГДК показників);
  - 2) перегляд інформації про ПС – перегляд короткої інформації про стан ПС на дату останнього спостереження із посиланням на створення звіту на основі даних моніторингу за певний період в таблично-графічному вигляді;
- перегляд карти розміщення постів спостереження:
- 1) перегляд інформації про ПС – перегляд короткої інформації про зареєстрований ПС із посиланням на створення звіту на основі даних моніторингу за певний період в таблично-графічному вигляді;
- формування звітності на основі даних моніторингу якості води – можна обрати один із доступних типів звітів, вказавши необхідні параметри для формування:
- 1) вибір періоду – вибір періоду моніторингу пункту спостереження;
  - 2) вибір показників – вибір показників для формування звітності у таблично-графічному вигляді;
  - 3) вибір водогосподарської організації;



- заповнення довідників – ведення інформаційного забезпечення системи;
- керування користувачами:
  - 1) реєстрація користувачів;
  - 2) управління користувачами в системі та надання їм певних прав.

## 1.2 Огляд наявних аналогів

Внаслідок огляду ринку програмних продуктів, які вирішують задачу моніторингу водних ресурсів, було виявлено:

- інформаційно-аналітичну систему – «Екологічний моніторинг»;
- інтерактивну карту забрудненості річок в Україні «Чиста вода»;
- WebPortal AQUARIUS;
- Environmental Monitoring System Software (EMS) - система моніторингу навколишнього середовища.

Система «Екологічний моніторинг» використовується для збору, аналітичної обробки та представлення інформації про стан навколишнього середовища регіону. В якості об'єктів дослідження і аналізу в системі задіяні атмосферне повітря, поверхневі та підземні води, ґрунти, об'єкти тваринного і рослинного світу. Система дозволяє проводити картографічну обробку і представлення даних, додавати та редагувати об'єкти. Наявне розмежування прав доступу і представлення інформації як спеціально уповноваженим особам, так і населенню [6].

Для спрощення сприйняття та аналізу інформації використовуються:

- розмальовка тематичної карти по точкам моніторингу;
- монотонні лінійні графіки по точкам моніторингу;
- кругові діаграми по вибраним точкам моніторингу.

Інтерактивна карта «Чиста вода» відображає дані Державного агентства водних ресурсів. На карті відображається понад 400 пунктів

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

контролю річкової води. Можна переглянути до 16 параметрів забруднення і дізнатися, як його рівень змінювався протягом 5 років [2].

WebPortal AQUARIUS – це рішення для надання онлайн-доступу в режимі реального часу до екологічних даних. Багата панель інструментів дозволяє отримати реальний знімок умов навколишнього середовища – наприклад, поточний стан води або якості повітря, місця затоплення, підвищення рівня резервуарів, тощо, завдяки сигналам від датчиків. Інтерактивна карта дозволяє збільшити потрібну область, визначити місце розташування та вивести повний список розрахованої статистики. Кольорові коди забезпечують візуальне відображення умов гідрології та якості води.

Сучасний графічний інтерфейс дозволяє швидко візуалізувати дані про воду через карти, сітки даних і діаграми. Відображається контекстна інформація. Наявна функція онлайн-сповіщень та автоматичних попереджень, що надсилаються електронною поштою або через SMS зацікавленим особам в разі виявлення незвичайних умов. Є можливість експорту даних, генерування звітності та розмежування прав доступу до системи [7].

Environmental Monitoring System Software (EMS) - це комерційна система збору, інтеграції та обробки даних метеорології, якості повітря та якості води. Система надає точні дані для відображення в режимі реального часу, надаючи установам найдосконаліші рішення для моніторингу довкілля [8]. Система має наступні можливості:

- моніторинг якості навколишнього середовища (якість повітря, якість води, гідрологічні дані);
- інтерактивні карти;
- формування спеціалізованих веб-звітів: щоденних, щомісячних, річних, загальні підсумки даних;
- візуалізація даних забруднення.

Як бачимо, всі розроблені програмні продукти вирішують завдання екологічної оцінки водних ресурсів, накопичення та відображення даних, побудови графіків та діаграм для зручного аналізу. Проте кожна з них має свої обмеження. Система «Екологічний моніторинг» розроблена багато років назад і вже давно не оновлювалася, тому користувацький інтерфейс є достатньо складним, а карта не дуже симпатична. Інтерактивна карта «Чиста вода» обмежена у своїй функціональності. Environmental Monitoring System Software (EMS) та WebPortal AQUARIUS є достатньо потужними системами, що отримують дані та здійснюють їх статистичну обробку в режимі реального часу завдяки встановленій системі датчиків. Проте ці програмні продукти мають дуже високу ціну та потребують значних зусиль та витрат для впровадження цих систем в Україні.

### 1.3 Постановка задачі

#### 1.3.1 Призначення розробки

Призначенням розробки є підтримка процесу проведення моніторингу та екологічної оцінки водних ресурсів України.

#### 1.3.2 Цілі та задачі розробки

Цілі розробки:

- покращення інформаційного супроводу робіт із здійснення моніторингу поверхневих вод та екологічної оцінки водних ресурсів України;
- підвищення оперативності та ефективності аналізу наявної інформації про стан водних ресурсів;
- спрощення регулярної актуалізації і багатоцільового використання інформації стосовно стану водних ресурсів.

Для досягнення поставлених цілей мають бути вирішені такі задачі:

- реєстрація пунктів спостереження;

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

- ведення моніторингу пунктів спостереження;
- можливість систематизації результатів довгострокових спостережень за станом водних ресурсів, його кількісними та якісними показниками;
- ведення контролю водогосподарської обстановки;
- створення карти перевищення ГДК показників на постах спостереження;
- створення карти розміщення пунктів спостереження;
- заповнення довідників;
- заповнення ГДК;
- внесення нормативів;
- формування звітності доступного типу за певний період;
- керування користувачами системи.

### Висновок до розділу

В даному розділі описано предметне середовище. Держводагенство є центральним органом, що отримує дані моніторингу від усіх підлеглих водогосподарських організацій та приймає рішення на основі отриманих результатів. Всі процеси та взаємодії, що відбуваються між водогосподарськими організаціями та Держводагенством описують структурні схеми опису діяльності, що наведені в частині графічного матеріалу. Процес роботи проілюстровано як до впровадження розробленої системи, так і після. Наведено функціональну модель і визначено головних акторів системи – адміністратор, співробітник, користувач. Для кожного варіанта використання наведено детальний опис.

Проаналізовано ринок наявних програмних продуктів, що вирішують задачу моніторингу водних ресурсів, і виявлено декілька аналогів. Після проведення порівняльного аналізу наявних аналогів було виявлено переваги розробки і необхідність її реалізації.

Розроблена система призначена для підтримки процесу проведення моніторингу та екологічної оцінки водних ресурсів України. Визначено основні цілі та задачі розробки, які мають бути вирішені для досягнення поставлених цілей.

## 2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

### 2.1 Вхідні дані

Всі вхідні дані вносять співробітники водогосподарських організацій та адміністратори системи. Розглянемо детальніше, які саме дані вносить кожен із представників цих двох груп.

**Дані, що надходять від адміністраторів.** Для того, щоб співробітники водогосподарських організацій мали можливість заходити в систему, адміністратори повинні спочатку зареєструвати їх і надати відповідні права доступу, заповнивши наступні дані:

- логін, пароль, роль;
- ПІБ, телефон, електронна пошта;
- установа, підрозділ, посада, дата призначення на посаду, офіс, територіальний код.

Для повноцінного користування системою необхідно заповнити наступні довідники відповідними значеннями:

- назва статусу пункту спостереження;
- назва району річкового басейну;
- назва головної лабораторії;
- назва водогосподарської організації;
- назва підрозділу водогосподарської організації;
- назва водного об'єкту I порядку;
- назва водного об'єкту II порядку;
- назва водного об'єкту III порядку;
- назва водного об'єкту IV порядку;
- назва водного об'єкту V порядку;
- назва типу нормативу;
- назва виду моніторингу;

- назва виду створу;
- назва періодичності контролю;
- назва ознаки транскордонних створів;
- назва групи показників;
- назва показника;
- назва лиману;
- назва області;
- назва типу;
- назва суббасейну;
- назва нормативу;
- назва групи;
- назва водосховища;
- назва водогосподарської системи;
- назва пункту контролю рівнів води;
- назва періоду;
- назва показника.

Для проведення моніторингу необхідно заповнити ГДК показників:

- група показників;
- показник;
- мінімальне та максимальне значення відповідно до нормативу.

Для контролю водогосподарської обстановки необхідно ввести відповідні нормативні значення:

- основний басейн, водосховище, проектний рівень, проектний обсяг, координати кривих об'ємів водосховища;
- найменування каналу, декадний план, сезони роботи із датами початку і кінця сезону;
- річка-пункт, щомісячні норми витрат;
- показник і його норма.

Дані, що надходять від співробітників водогосподарських організацій. Для того, щоб розпочати процес моніторингу рівня забрудненості водних об'єктів, треба для початку зареєструвати пост спостереження і внести наступні дані:

- область, район річкового басейну, водний об'єкт І категорії, статус ПС;
- опис місця розташування ПС, відстань до лабораторії, населений пункт, найближчий до створу, суббасейн, лиман, назва головної лабораторії, назва лабораторії-партнера, назва водогосподарської організації, назва підрозділу водогосподарської організації, організація, що відповідає за відбір проб, водний об'єкт І-V категорії, відомчий код ПС, код водогосподарської ділянки, вид створу, тип нормативу, вид моніторингу, періодичність контролю, транскордонний створ з якою країною, групи показників, географічні координати, ознака відображення на головній карті.

Після реєстрації пункту спостереження можна переходити до моніторингу. Для цього необхідно заповнити наступні дані:

- пункт спостереження;
- виміряні значення показників.

Також система надає можливість контролю водогосподарської обстановки. Для контролю гідрологічної ситуації на водосховищах вносяться наступні дані:

- дата;
- водоймище;
- фактичний рівень, фактичний приплив, фактичний скид;
- коментар.

Також є можливість завантаження даних із файлу заданої структури.

Для контролю водогосподарських систем вносять значення середньодобового збору.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		19



Для контролю гідрологічної ситуації на контрольних пунктів вносять такі дані:

- річка-пункт;
- фактичний рівень, фактичні витрати.

Для контролю водогосподарської обстановки вносять наступні дані:

- річка-пункт;
- показник і фактичне значення.

## 2.2 Вихідні дані

На основі даних проведеного моніторингу для подальшого аналізу отриманих результатів будуються наступні звіти:

- реєстр ПС:
  - 1) кількість ПС по Головним річковим басейнам;
  - 2) кількість ПС по Областям;
  - 3) пости спостереження;
- дані моніторингу:
  - 1) дані моніторингу (за адміністративно-територіальним принципом);
  - 2) дані моніторингу (за видом створу);
  - 3) дані моніторингу (за водогосподарською організацією);
  - 4) дані моніторингу (за одним показником), таблично-графічне представлення даних;
  - 5) дані моніторингу (за ознакою транскордонного створу);
  - 6) дані моніторингу (за районом річкового басейну або суббасейну);
  - 7) дані моніторингу (по руслу Нижнього Дніпра);
  - 8) дані моніторингу (по руслу Нижнього Дніпра), графічне представлення даних за одним показником;

- 9) дані моніторингу (по руслу Нижнього Дніпра), таблично-графічне представлення даних за групою показників;
- 10) дані моніторингу (по руслу р. Десна);
- 11) дані моніторингу (по руслу р. Десна), графічне представлення даних за одним показником;
- 12) дані моніторингу (по руслу р. Десна), таблично-графічне представлення даних за групою показників;
- 13) дані моніторингу (по руслу р. Дніпро);
- 14) дані моніторингу (по руслу р. Дніпро), графічне представлення даних за одним показником;
- 15) дані моніторингу (по руслу р. Дніпро), таблично-графічне представлення даних за групою показників;
- 16) дані моніторингу (по руслу р. Дністер);
- 17) дані моніторингу (по руслу р. Дністер), графічне представлення даних за одним показником;
- 18) дані моніторингу (по руслу р. Дністер), таблично-графічне представлення даних за групою показників;
- 19) дані моніторингу (по руслу р. Дунай);
- 20) дані моніторингу (по руслу р. Дунай), графічне представлення даних за одним показником;
- 21) дані моніторингу (по руслу р. Дунай), таблично-графічне представлення даних за групою показників;
- 22) дані моніторингу (по руслу р. Західний Буг);
- 23) дані моніторингу (по руслу р. Західний Буг), графічне представлення даних за одним показником;
- 24) дані моніторингу (по руслу р. Західний Буг), таблично-графічне представлення даних за групою показників;
- 25) дані моніторингу (по руслу р. Південний Буг);

- 26) дані моніторингу (по руслу р. Південний Буг), графічне представлення даних за одним показником;
- 27) дані моніторингу (по руслу р. Південний Буг), таблично-графічне представлення даних за групою показників;
- 28) дані моніторингу (по руслу р. Прип'ять);
- 29) дані моніторингу (по руслу р. Прип'ять), графічне представлення даних за одним показником;
- 30) дані моніторингу (по руслу р. Прип'ять), таблично-графічне представлення даних за групою показників;
- 31) дані моніторингу (по руслу р. Прут);
- 32) дані моніторингу (по руслу р. Прут), графічне представлення даних за одним показником;
- 33) дані моніторингу (по руслу р. Прут), таблично-графічне представлення даних за групою показників;
- 34) дані моніторингу (по руслу р. Сіверський Донець);
- 35) дані моніторингу (по руслу р. Сіверський Донець), графічне представлення даних за одним показником;
- 36) дані моніторингу (по руслу р. Сіверський Донець), таблично-графічне представлення даних за групою показників;
- 37) дані моніторингу (по руслу р. Сірет);
- 38) дані моніторингу (по руслу р. Сірет), графічне представлення даних за одним показником;
- 39) дані моніторингу (по руслу р. Сірет), таблично-графічне представлення даних за групою показників;
- 40) дані моніторингу (по руслу р. Тиса);
- 41) дані моніторингу (по руслу р. Тиса), графічне представлення даних за одним показником;
- 42) дані моніторингу (по руслу р. Тиса), таблично-графічне представлення даних за групою показників;

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

- 43) дані моніторингу (по руслу Середнього Дніпра);
- 44) дані моніторингу (по руслу Середнього Дніпра), графічне представлення даних за одним показником;
- 45) дані моніторингу (по руслу Середнього Дніпра), таблично-графічне представлення даних за групою показників;
- загальні дані моніторингу:
  - 1) моніторинг за даними поста спостереження (10 показників), таблично-графічне представлення даних;
  - 2) моніторинг за даними поста спостереження (всі показники);
- інші:
  - 1) дані моніторингу (за водогосподарською організацією);
  - 2) моніторинг за даними поста спостереження (10 показників), середньорічні значення;
  - 3) середні значення показників (за районом річкового басейну або суббасейну);
- звіти загального користування:
  - 1) дані моніторингу (за адміністративно-територіальним принципом);
  - 2) дані моніторингу (за водогосподарською організацією);
  - 3) дані моніторингу (за ознакою транскордонного створу);
  - 4) дані моніторингу (за районом річкового басейну або суббасейну);
- звіт «Інформація про водогосподарську обстановку в Україні».

У таблицях 2.1-2.7 наведено макети деяких звітів.

Таблиця 2.1 – Макет звіту «Кількість ПС по головним річковим басейнам»

Номер	Назва головного річкового басейну	Кількість постів спостереження

Таблиця 2.2 – Макет звіту «Пости спостереження»

Номер	Назва ПС	Відомчий код ПС	Географічні координати	
			Широта	Довгота

Таблиця 2.3 – Макет звітів групи «Загальні дані моніторингу. Моніторинг за даними поста спостереження»

Дата	Амоні й- іони, мг/дм 3	Біохімі чне спожив анн я кисн ю за 5 діб, мгО2/д м3	Завис лі (суспе ндов ані) речов ини, мг/дм 3	Кисе нь розч инен и й, мг О2/д м3	Нітра т- іони, мг/дм 3	Нітр ит- іони, мг/д м3	Суль фат- іони, мг/дм 3	Фосфа т- іони (поліф осф ати), мг/дм3	Хлор ид- іони, мг/дм 3

Таблиця 2.4 – Макет першої таблиці звіту «Інформація про водогосподарську обстановку в Україні»

Водоймище	Проектні дані		Фактичні дані				Вільний обсяг, млн.м3
	Рівень, м	Обсяг, млн.м3	Рівень, м	Обсяг, млн.м3	Приток, м3/с	Скид, м3/с	

Таблиця 2.5 – Макет другої таблиці «Водогосподарські системи» звіту  
«Інформація про водогосподарську обстановку в Україні»

Найменування каналів, зрошувальних систем	Забор води з початку сезону, тис.м3	Декадний план, тис.м3	Забор води з початку декади, тис.м3	Середньодобовий забор води, м3/сек

Таблиця 2.6 – Макет третьої таблиці «Гідрологічна обстановка» звіту  
«Інформація про водогосподарську обстановку в Україні»

Річка-Пункт	Рівень, м			Витрати, м3/сек	
	MAX	MIN	Факт	Норма	Факт

Таблиця 2.7 – Макет четвертої таблиці «Гідрохімічна обстановка» звіту  
«Інформація про водогосподарську обстановку в Україні»

Річка-Пункт	Загальна мінералізація, мг/л		Хлориди, мг/л		Жорсткість, мг.екв/л	
	Норма	Факт	Норма	Факт	Норма	Факт

Також вихідними даними є обчислені значення перевищення ГДК вимірюваних показників на постах спостереження для моніторингу рівня

забрудненості водних об'єктів, що візуалізуються на карті відповідними кольоровими індикаторами. По кожному посту спостереження можна переглянути коротку інформацію про пост та значення показників на останню дату спостереження у порівнянні з ГДК.

### 2.3 Опис структури бази даних

У проекті використовуються 2 бази даних:

- Admin\_DB – база даних для збереження службових даних про користувачів та їх ролі;
- WaterMon\_DB – основна база даних для збереження всіх даних моніторингу та водогосподарської обстановки.

Схеми структурні баз даних Admin\_DB, WaterMon\_DB наведені в частині графічного матеріалу.

У таблицях 2.1 – 2.7 наведено опис таблиць бази даних Admin\_DB.

Таблиця 2.1 – Опис таблиці modules

Код	Опис	Тип даних
id	Ідентифікатор плагіну	int
code	Код плагіну	nvarchar(30)
blocked	Ознака неактивності плагіну	bit

Таблиця 2.2 – Опис таблиці UserRoleFunctions

Код	Опис	Тип даних
id	Ідентифікатор запису	int
group_name	Назва групи користувачів	nvarchar(100)
district_type	Тип району	int
func_name	Ім'я функції	nvarchar(max)

Таблиця 2.3 – Опис таблиці Users

Код	Опис	Тип даних
UserId	Ідентифікатор користувача	uniqueidentifier
UserName	Ім'я користувача	nvarchar(256)
LoweredUserName	Ім'я користувача в нижньому регістрі	nvarchar(256)
IsAnonymous	Ознака анонімності	bit
LastActivityDate	Остання дата активності	datetime

Таблиця 2.4 – Опис таблиці Profile

Код	Опис	Тип даних
UserId	Ідентифікатор користувача	uniqueidentifier
PropertyNames	Назви атрибутів користувача	ntext
PropertyValuesString	Значення атрибутів користувача	ntext
PropertyValuesBinary	Значення атрибутів користувача в бінарному вигляді	image
LastUpdatedDate	Остання дата активності	datetime

Таблиця 2.5 – Опис таблиці Roles

Код	Опис	Тип даних
RoleId	Ідентифікатор ролі	uniqueidentifier
RoleName	Назва ролі	nvarchar(256)
LoweredRoleName	Назва ролі в нижньому регістрі	nvarchar(256)



Продовження таблиці 2.5

Код	Опис	Тип даних
Description	Опис ролі	nvarchar(256)

Таблиця 2.6 – UsersInRoles

Код	Опис	Тип даних
UserId	Ідентифікатор користувача	uniqueidentifier
RoleId	Ідентифікатор ролі	uniqueidentifier

Таблиця 2.7 – Опис таблиці UserExtraProperties

Код	Опис	Тип даних
id	Ідентифікатор	int
UserId	Ідентифікатор користувача	uniqueidentifier
Code	Код атрибута	nvarchar(50)
Value	Значення	nvarchar(1000)

У таблицях 2.8 – 2.26 наведено опис таблиць бази даних WaterMon\_DB.

Таблиця 2.8 – Опис таблиці LastDatePostData

Код	Опис	Тип даних
IDCard_Reestr	Ід запису зареєстрованого поста спостереження	numeric(18,0)
IDCard_Monitoring	Ід запису про моніторинг показників забруднення поста спостереження	numeric(18,0)

Продовження таблиці 2.8

Код	Опис	Тип даних
Post_Name	Назва поста спостереження	nvarchar(255)
Riverbas_Name	Назва басейну	nvarchar(50)
WaterLab_Name	Назва лабораторії	nvarchar(255)
Latitude	Координата довготи ПС	numeric(18,6)
Longitude	Координата широти ПС	numeric(18,6)
Controle_Date	Дата спостережень	Date
Azot_Value	Азот загальний, фактичне значення показника	float
Azot_Norm	Азот загальний, ГДК	float
Azot_Excess	Азот загальний, перевищення ГДК	float
BSK5_Value	Біологічне споживання кисню за 5 діб, фактичне значення	float
BSK5_Norm	Біологічне споживання кисню за 5 діб, ГДК	float
BSK5_Excess	Біологічне споживання кисню за 5 діб, перевищення ГДК	float
Zavisli_Value	Завислі речовини, фактичне значення	float
Zavisli_Norm	Завислі речовини, ГДК	float
Zavisli_Excess	Завислі речовини, перевищення	float

Продовження таблиці 2.8

Код	Опис	Тип даних
Kisen_Value	Кисень розчинений, фактичне значення	float
Kisen_Norm	Кисень розчинений, ГДК	float
Kisen_Excess	Кисень розчинений, перевищення ГДК	float
Sulfat_Value	Сульфат-іони, фактичне значення	float
Sulfat_Norm	Сульфат-іони, ГДК	float
Sulfat_Excess	Сульфат-іони, перевищення ГДК	float
Hlorid_Value	Хлорид-іони, фактичне значення	float
Hlorid_Norm	Хлорид-іони, ГДК	float
Hlorid_Excess	Хлорид-іони, перевищення ГДК	float
Amoniy_Value	Амоній-іони, фактичне значення	float
Amoniy_Norm	Амоній-іони, ГДК	float
Amoniy_Excess	Амоній-іони, перевищення ГДК	float
Nitrat_Value	Нітрат-іони, фактичне значення	float
Nitrat_Norm	Нітрат-іони, ГДК	float
Nitrat_Excess	Нітрат-іони, перевищення ГДК	float

Продовження таблиці 2.8

Код	Опис	Тип даних
Nitrit_Value	Нітрит-іони, фактичне значення	Float
Nitrit_Norm	Нітрит-іони, ГДК	float
Nitrit_Excess	Нітрит-іони, перевищення ГДК	float
Fosfat_Value	Фосфат-іони, фактичне значення	float
Fosfat_Norm	Фосфат-іони, ГДК	float
Fosfat_Excess	Фосфат-іони, перевищення ГДК	float
Count_Attribute_0	Кількість атрибутів, що не перевищують ГДК	int
Count_Attribute_1-1.5	Кількість показників, перевищення ГДК яких у межах 1-1,5	Int
Count_Attribute_1.5-2.0	Кількість показників, перевищення ГДК яких у межах 1,5-2	Int
Count_Attribute_2.0-2.5	Кількість показників, перевищення ГДК яких у межах 2-2,5	Int
Count_Attribute_2.5	Кількість показників, перевищення ГДК яких більше 2,5	Int

Продовження таблиці 2.8

Код	Опис	Тип даних
Color_Index	Індекс кольору індикатора для відображення на карті	Int
Diameter_Index	Індекс діаметру індикатора	Int
Total_Index	Тотальний індекс – індекс кольору+індекс діаметру	nvarchar(10)
Report_ID	Ід шаблону звіту	Int

Таблиця 2.9 – Опис таблиці WaterMon\_Value

Код	Опис	Тип даних
DVL_IDP	Ідентифікатор запису	numeric(18,0)
DVL_USR	Ідентифікатор користувача, який зберіг дані	nvarchar(50)
DVL_DAT	Дата створення запису	datetime
DLS_IDP	Ідентифікатор розділу	numeric(18,0)
DVL_TEXT1	Поле для збереження текстових значень	nvarchar(max)
...	...	...
DVL_TEXT70	Поле для збереження текстових значень	nvarchar(max)
DVL_DATE1	Поле для збереження дат	datetime
...	...	...

Продовження таблиці 2.9

Код	Опис	Тип даних
DVL_DATE25	Поле для збереження дат	datetime
DVL_NUM1	Поле для збереження чисел	numeric(18,6)
...	...	...
DVL_NUM120	Поле для збереження чисел	numeric(18,6)
DVL_FILE1	Поле для збереження бінарних даних	varbinary(max)
...	...	...
DVL_FILE10	Поле для збереження бінарних даних	varbinary(max)
DVL_CTXT1	Поле для збереження бінарних даних	varbinary(256)
...	...	...
DVL_CTXT15	Поле для збереження бінарних даних	varbinary(256)

Таблиця 2.10 – Опис таблиці Client

Код	Опис	Тип даних
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
CLI_PIDP	Ідентифікатор батьківського клієнта	numeric(18,0)
CLI_NAME	Назва клієнта	nvarchar(255)
CLI_FNAME	Повна назва клієнта	nvarchar(500)
CLI_DDAT	Дата завершення актуальності запису	datetime

Таблиця 2.11 – Опис таблиці Client\_Local

Код	Опис	Тип даних
LCL_IDP	Ідентифікатор запису	numeric(18,0)
LLC_IDP	Ідентифікатор мови	numeric(18,0)
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
LCL_FNAME	Повна назва клієнта	nvarchar(1024)
LCL_SNAME	Коротка назва клієнта	nvarchar(1024)
LCL_LATN	Повна назва клієнта(латиницею)	nvarchar(1024)
LCL_CCOU	Код країни	int
LCL_NCOU	Назва країни	nvarchar(100)
LCL_ADR	Адреса	nvarchar(1024)
LCL_CITY	Місто	nvarchar(35)
LCL_STAT	Штат(регіон)	nvarchar(50)
LCL_PCOD	Поштовий код	nvarchar(15)
LCL_TEL	Телефон	nvarchar(10)
LCL_ATEL	Додаток до телефону	nvarchar(5)
LCL_CTEL	Код країни для телефону	nvarchar(3)
LCL_FAX	Факс	nvarchar(10)
LCL_MAIL	Електронна пошта	nvarchar(100)

Таблиця 2.12 – Опис таблиці Dictionary\_Client\_Link

Код	Опис	Тип даних
CDL_IDP	Ідентифікатор запису	numeric(18,0)
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
DIC_IDP	Ідентифікатор довідника	numeric(18,0)
DIC_UID	Код довідника	nvarchar(1024)

Таблиця 2.13 – Опис таблиці DictionaryValue\_Client\_Link

Код	Опис	Тип даних
CDVL_IDP	Ідентифікатор запису	numeric(18,0)
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
DICV_IDP	Ідентифікатор значення довідника	numeric(18,0)
DICV_UID	Код значення довідника	nvarchar(1024)

Таблиця 2.14 – Опис таблиці Type\_Attribute

Код	Опис	Тип даних
TYP_IDP	Ідентифікатор запису	numeric(18,0)
DAT_IDP	Ідентифікатор атрибуту	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
TPE_IDP	Ідентифікатор назви типу	numeric(18,0)
TYP_REQ	Обов'язковість атрибуту (0-ні, 1-так)	int
TYP_DIM	Розмірність значення атрибуту	int
TYP_DFR	Розмірність дробової частини	int
TYP_REG	Регулярний вираз	nvarchar(max)
TYP_MES	Текстове повідомлення	nvarchar(100)
TYP_CLV	Ширина стовпчика ґріда для значення атрибуту	int
TYP_DIS	Ознака відображення в ґріді	int



Продовження таблиці 2.14

Код	Опис	Тип даних
TYP_AID	Ідентифікатор атрибуту зв'язаного досьє	nvarchar(255)
TYP_SRT	Ознака сортування (0-відсутнє, 1-за зростанням, 2-за спаданням)	int
TYP_ORD	Порядок сортування	int
TYP_CSC	Код використання в контекстному пошуку	int
TYP_CSO	Порядок сортування в контекстному пошуку	int
TYP_TAS	Ознака технічного поля	int
TYP_UFS	Ознака відображення в універсальному фільтрі	int

Таблиця 2.15 – Опис таблиці Dictionary

Код	Опис	Тип даних
DIC_IDP	Ідентифікатор запису	numeric(18,0)
DIC_TREE	Рівень ієрархії довідника	numeric(18,0)
DIC_LID	Зв'язок атрибута з довідником	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
DIC_NAME	Назва	nvarchar(255)
DIC_UID	Унікальний ідентифікатор	numeric(18,0)
DIC_ALL	Для всіх	bit

Таблиця 2.16 – Опис таблиці Dictionary\_Value

Код	Опис	Тип даних
DICV_IDP	Ідентифікатор запису	numeric(18,0)
DIC_IDP	Зв'язок із довідником	numeric(18,0)
DICV_LID	Зв'язок з головним атрибутом	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
DICV_LNAME	Повна назва атрибуту	nvarchar(max)
DICV_SNAME	Коротка назва атрибуту	nvarchar(max)
DICV_SDAT	Дата створення запису	datetime
DICV_EDAT	Дата закриття запису	datetime
DICV_LVER	Попередня версія	numeric(18,0)
DICV_UID	Унікальний ідентифікатор атрибуту	numeric(18,0)
DICV_SORT	Порядок сортування	numeric(18,0)
DICV_CODE	Локальний код атрибуту	nvarchar(100)

Таблиця 2.17 – Опис таблиці WaterMon\_Header

Код	Опис	Тип даних
DOS_IDP	Ідентифікатор запису	numeric(18,0)
DOS_LID	Зв'язок досьє з розділом	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
DOS_LNAM	Повна назва досьє або розділу	nvarchar(500)
DOS_SNAM	Коротка назва досьє або розділу	nvarchar(255)
DOS_UID	Унікальний ідентифікатор атрибута	numeric(18,0)

Продовження таблиці 2.17

Код	Опис	Тип даних
DOS_TYP	Тип категорії досьє (NULL-досьє, 1 – досьє- довідник)	int
DOS_NUM	Порядковий номер вкладки (1-головна)	int
MOD_IDP	Ідентифікатор модуля	int

Таблиця 2.18 – Опис таблиці WaterMon\_Attribute

Код	Опис	Тип даних
DAT_IDP	Ідентифікатор запису	numeric(18,0)
DAT_LID	Зв'язок атрибутів	numeric(18,0)
DAT_LDOS	Ідентифікатор досьє- довідника	numeric(18,0)
DOS_IDP	Ідентифікатор розділу досьє	numeric(18,0)
DIC_IDP	Ідентифікатор атрибуту довідника	numeric(18,0)
DAT_GID	Ідентифікатор батьківського ґріда	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
DAT_LNAM	Повна назва атрибуту	nvarchar(500)
DAT_SNAM	Коротка назва атрибуту	nvarchar(255)
DAT_FNAM	Назва поля таблиці даних	nvarchar(50)
DAT_UID	Унікальний ідентифікатор атрибуту	numeric(18,0)

Таблиця 2.19 – Опис таблиці WaterMon\_Buttons

Код	Опис	Тип даних
DOB_IDP	Ідентифікатор запису	numeric(18,0)
DOS_IDP	Ідентифікатор досьє	numeric(18,0)
DOB_NAME	Назва кнопки	nvarchar(128)
DOB_TYP	Тип кнопки	nvarchar(20)
DOB_POS	Положення кнопки	nvarchar(1)
DOB_ORD	Порядок кнопки	int
DOB_HTML	Відображення	nvarchar(max)
DOB_SCR	Скрипт опрацювання (onClick)	nvarchar(max)

Таблиця 2.20 – Опис таблиці WaterMon\_Item

Код	Опис	Тип даних
DLS_IDP	Ідентифікатор запису	numeric(18,0)
DLS_DID	Ідентифікатор досьє	numeric(18,0)
DLS_PID	Ідентифікатор розділу досьє	numeric(18,0)
DLS_LID	Зв'язок розділу з досьє	numeric(18,0)
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
DLS_CDAT	Дата створення досьє	datetime
DLS_DDAT	Дата видалення досьє	datetime
DLS_DAUT	Автор досьє	nvarchar(50)
DLS_EDIT	Ознака редагування	int
DLS_EDAT	Дата початку редагування	datetime
DLS_NEDI	Редактор (логін користувача)	nvarchar(50)
LLS_IDP	Ідентифікатор мови	numeric(18,0)

Таблиця 2.21 – Опис таблиці Item\_Department\_Link

Код	Опис	Тип даних
DTL_IDP	Ідентифікатор запису	numeric(18,0)
DLS_IDP	Ідентифікатор досьє	numeric(18,0)
DTL_STS	Статус документа	numeric(18,0)
DTL_DAT	Дата зміни статусу	datetime
DTL_ACT	Актуальність запису	int
DTL_AUT	Автор зміни статусу	uniqueidentifier
STT_UID	Ідентифікатор підрозділу	numeric(18,0)

Таблиця 2.22 – Опис таблиці WaterMon\_GridValue

Код	Опис	Тип даних
GRD_IDP	Ідентифікатор запису	numeric(18,0)
DVL_IDP	Ідентифікатор значення досьє	numeric(18,0)
GRD_LID	Ідентифікатор назви атрибуту ґріда	numeric(18,0)
GRD_NUM	Номер за порядком	int
GRD_TEXT1	Поле для збереження тексту	nvarchar(max)
...		
GRD_TEXT30	Поле для збереження тексту	nvarchar(max)
GRD_DATE1	Поле для збереження дат	datetime
...		

Продовження таблиці 2.22

Код	Опис	Тип даних
GRD_DATE10	Поле для збереження дат	datetime
GRD_NUM1	Поле для збереження чисел	numeric(18,6)
...		
GRD_NUM20	Поле для збереження чисел	numeric(18,6)
GRD_FILE1	Поле для збереження файлів	varbinary(max)
...		
GRD_FILE10	Поле для збереження файлів	varbinary(max)
GRD_PIDP	Ідентифікатор батьківського запису	numeric(18,0)
GRD_CTXT1	Поле для збереження бінарних даних	varbinary(256)
...		
GRD_CTXT15	Поле для збереження бінарних даних	varbinary(256)

Таблиця 2.23 – Опис таблиці Type\_Data

Код	Опис	Тип даних
TPE_IDP	Ідентифікатор запису	numeric(18,0)
TPE_NAME	Назва типу	nvarchar(100)

Таблиця 2.24 – Опис таблиці Agency\_Structure

Код	Опис	Тип даних
STT_IDP	Ідентифікатор запису	numeric(18,0)
STT_LID	Ідентифікатор головного елемента дерева	numeric(18,0)
LLS_IDP	Ідентифікатор мови	numeric(18,0)
DICV_IDP	Ідентифікатор посади співробітника	numeric(18,0)
CLI_IDP	Ідентифікатор клієнта	numeric(18,0)
STT_DSV	Дата створення елемента	datetime
STT_DDA	Дата зняття актуальності	datetime
STT_USID	Ідентифікатор користувача в системі	uniqueidentifier
STT_LOG	Логін співробітника в системі	nvarchar(100)
STT_TYP	Тип запису (1-підрозділ, 2 - співробітник)	int
STT_LNAM	Повна назва (ПІБ співробітника)	nvarchar(250)
STT_SNAM	Коротка назва	nvarchar(100)
STT_SURN	Прізвище співробітника	nvarchar(100)
STT_FIRN	Ім'я співробітника	nvarchar(100)
STT_MIDN	По-батькові	nvarchar(100)
STT_DST	Дата призначення	datetime
STT_CHI	Керівник	nvarchar(255)

Продовження таблиці 2.24

Код	Опис	Тип даних
STT_ABR	Абревіатура	nvarchar(50)
STT_TEL	Телефон	nvarchar(50)
STT_MAI	Електронна пошта	nvarchar(50)
STT_ROO	Офіс	nvarchar(50)
STT_LEV	Рівень ієрархії	int
STT_NUM	Номер за порядком	int
STT_COM	Коментар	nvarchar(4000)
STT_UID	Унікальний ідентифікатор значення	numeric(18,0)
STT_CODE	Локальний код	nvarchar(50)

Таблиця 2.25 – Опис таблиці Language\_List

Код	Опис	Тип даних
LLS_IDP	Ідентифікатор запису	numeric(18,0)
LLS_NAME	Назва мови	nvarchar(50)
LLS_CODE	Код мови	nvarchar(3)
LLS_UCODE	Код мови в нижньому регістрі	varchar(max)

Таблиця 2.26 – Опис таблиці Reports

Код	Опис	Тип даних
id	Ідентифікатор запису	int
number	Номер звіту	nvarchar(20)
name	Назва звіту	nvarchar(250)
sql_query	Запит	nvarchar(max)
template_name	Назва шаблону	nvarchar(250)



Продовження таблиці 2.26

Код	Опис	Тип даних
Template	Шаблон	varbinary(max)
input_params	Вхідні параметри	nvarchar(max)
category_id	Категорія звіту	int
display_script	Скрипт	nvarchar(max)
data_class_name	Назва класу контекстного об'єкта скрипта	nvarchar(255)
Date_changes	Дата редагування	int
UserName	Редактор (логін користувача)	nvarchar(256)

### Висновок до розділу

У цьому розділі описано інформаційне забезпечення системи. Наведено всі вхідні дані, що надходять від адміністраторів та співробітників водогосподарських організацій. Адміністратори вносять службові дані, які необхідні для забезпечення повного функціонування системи. Водогосподарські організації вносять всі дані, що стосуються моніторингу водних об'єктів та ведення водогосподарської обстановки. Визначено повний перелік категорій звітів, що складає вихідні дані. Вихідними даними також є графічне відображення об'єктів моніторингу на карті перевищення ГДК показників на постах спостереження та на карті розміщення постів спостереження.

Представлено структурні схеми баз даних Admin\_DB та WaterMon\_DB, що використовуються відповідно для збереження службових даних та всіх даних, що стосуються моніторингу та водогосподарської обстановки. Наведено повний опис таблиць відповідних баз даних.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		45

### 3 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

#### 3.1 Змістовна постановка задачі

Одним із завдань Держводагентства є контроль стану водойм України. Для цього реєструються пости спостереження і з певною періодичністю вимірюються різні показники водойм, такі як амоній-іони, завислі речовини, кисень розчинений та інші, що дозволяють оцінювати поточний стан об'єкта. Вимірювання показників здійснюється на протязі більше, ніж 20-ти років, що дозволяє спостерігати динаміку процесу зміни значень показників та робити певні прогнози на майбутнє. Необхідно для поста спостереження р.Дніпро, м.Київ, Гідропарк на основі історичних даних за період 1993-2018рр. спрогнозувати значення показника амоній іони на наступний рік. Саме наявність іонів амонію в концентраціях, що перевищують допустиму, вказує на забруднення і близькість джерела забруднення (комунальні очисні споруди, промислові відходи, ферми, азотні добрива та ін.)

#### 3.2 Математична постановка задачі

Нехай на пості спостереження р.Дніпро, м.Київ, гідропарк значення показника амоній іони виміряні в дискретні моменти часу  $t = 1, 2, 3, \dots, T$ . Виміряні значення показника амоній іони за період  $T$  формують часовий ряд  $Y_1, Y_2, \dots, Y_T$ . В момент часу  $T$  необхідно спрогнозувати значення показника  $Y_t$  в момент часу  $t = T + 1$ .

#### 3.3 Обґрунтування методу розв'язання

Для розв'язання поставленої задачі прогнозування використовуються методи екстраполяції [9]. Екстраполяційні методи є дуже поширеними і найбільш розробленими формалізованими методами прогнозування.

Екстраполяція – це метод, що базується на поширенні минулих і

теперішніх тенденцій і закономірностей на подальший розвиток об'єкта прогнозування. Мета такого прогнозу – показати, до яких результатів можна дійти в майбутньому, якщо рухатися до нього з тією самою швидкістю або прискоренням, що й у минулому.

Перевагою методу прогнозної екстраполяції є зв'язок минулого, теперішнього і майбутнього, формалізація цього зв'язку. Цей метод дозволяє усереднити вплив одиничних врахованих і загальний вплив неврахованих факторів.

Існують різні прийоми екстраполяції. Ми будемо використовувати найпопулярніші методи – метод ковзного середнього, метод експоненціального згладжування, метод найменших квадратів.

Для вирішення задачі прогнозування також використаємо такий сучасний метод як нейромережеві технології. Нейронні мережі [10] доцільно використовувати для вирішення погано формалізованих завдань, які вимагають трудомістких обчислень. Саме до такого класу задач належить задача прогнозування. Здатність штучних нейронних мереж до узагальнення і виявлення прихованих залежностей усередині елементів мережі дозволяє вирішити поставлену задачу.

### 3.4 Опис методів розв'язання

#### 3.4.1 Методи апроксимації

**Метод ковзного середнього** [11] є звичайним середнім арифметичним від значення показника за певний період. Ковзне середнє являє собою якийсь показник рівноваги за певний період. Усереднюючи значення показника, ковзне середнє завжди слідує за головною тенденцією динаміки змін, фільтруючи дрібні коливання. Узагальнимо описаний вище метод у вигляді покрокового алгоритму:

**КРОК 1.** Визначити величину інтервалу згладжування рівною 3.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

**КРОК 2.** Розрахувати прогнозоване значення показника як середнє арифметичне значень показника за попередні 3 періоди за формулою

$$Y_{t+1} = \frac{(Y_{t-2} + Y_{t-1} + Y_t)}{3}, t = 3, 4, \dots, T \quad (3.1)$$

При розрахунку прогнозного значення **методом експоненціального згладжування** враховується відхилення попереднього прогнозу від реального показника, а сам розрахунок проводиться за формулою

$$Y_{t+1} = Y'_t + \alpha \times (Y_t - Y'_t), t = 1, 2, \dots, T, \quad (3.2)$$

де  $Y'_{t+1}$  – наступне значення показника, що прогнозується;

$Y'_t$  – попереднє значення показника, що було спрогнозоване;

$\alpha$  – параметр згладжування ( $0 < \alpha < 1$ );

$Y_t$  – фактичне значення показника.

Від величини  $\alpha$  залежить, як швидко зменшується вага впливу попередніх спостережень. Узагальнимо описаний вище метод у вигляді покрокового алгоритму:

**КРОК 1.** Задати значення параметра згладжування  $\alpha = 0,3$ .

**КРОК 2.** Задати значення  $Y'_1 = Y_1$ .

**КРОК 3.** Розрахувати прогнозоване значення показника  $Y_{t+1}$  за формулою 3.2.

Суть **методу найменших квадратів** [12] полягає в мінімізації суми квадратичних відхилень між спостережуваними та розрахованими величинами. Згладжування часового ряду методом найменших квадратів служить для відображення закономірності поведінки досліджуваного показника. В якості типу кривої оберемо лінійну залежність.

Отже, робоча формула методу

$$Y_t = a \times t + b, \quad (3.3)$$

де  $Y_t$  – прогнозований показник ;

$t$  – прогнозований період;

$a, b$  – коефіцієнти.

Розрахунок  $a$  і  $b$  відбувається за наступними формулами

$$a = \frac{\sum_{i=1}^n (Y_f \times X) - \frac{\sum_{i=1}^n X \times \sum_{i=1}^n Y_f}{n}}{\sum_{i=1}^n X^2 - \frac{(\sum_{i=1}^n X)^2}{n}}, \quad (3.4)$$

$$b = \frac{\sum_{i=1}^n Y_f}{n} - \frac{a \times \sum_{i=1}^n X}{n} \quad (3.5)$$

де  $X$  – порядковий номер елемента динаміки;

$Y_f$  – фактичне значення ряду динаміки;

$n$  – кількість рівнів часового ряду.

### 3.4.2 Нейронна мережа

Розглянемо метод прогнозування з використанням штучної нейронної мережі.

Найголовнішим питанням при реалізації нейронної мережі є моделювання архітектури. Це питання вибору кількості прихованих шарів, кількість нейронів для кожного прихованого шару, способи з'єднання цих нейронів, а також вибір алгоритму навчання. Типи архітектур нейронних мереж поділяють на 2 великі класи: мережі прямого розповсюдження і мережі із зворотніми зв'язками (рекурентні). До першого класу належать перцептрони.

На практиці найчастіше використовуються саме багат шарові перцептрони. Перцептрон має просту структуру і є достатньо універсальним. Багат шарові перцептрони використовують для розв'язання задач апроксимації, прогнозування, класифікації тощо. Простота і універсальність перцептрона є основною причиною вибору його в якості основи.

Отже, прогнозування значення показника забруднення водойм буде реалізовано на основі багатошарового персептрона.

Кількість нейронів на вхідному шарі рівне кількості факторів, що впливають на значення показника – у нас це попередні значення показника – і складає 5 нейронів. Так як необхідно здійснити прогноз однієї величини – значення показника, то будемо використовувати лише 1 нейрон у вихідному шарі.

Немає конкретних правил для вибору кількості прихованих шарів та кількості нейронів в них, проте існують обмеження, що допомагають зробити вибір [13]. Оскільки функція визначена на скінченній множині точок, то 3-шаровий персептрон здатний її апроксимувати.

Для вибору кількості нейронів в прихованих шарах існують евристичні правила. Одним із таких правил є правило геометричної піраміди. По цьому правилу кількість нейронів в прихованому шарі 3-шарового персептрона обчислюється за наступною формулою:

$$k = \sqrt{n \times m}, \quad (3.6)$$

де  $k$  – кількість нейронів в прихованому шарі;

$n$  – кількість нейронів у вхідному шарі;

$m$  – кількість нейронів у вихідному шарі.

Отже, кількість нейронів у прихованому шарі 3.

Структура нейронної мережі зображена на рисунку 3.1.

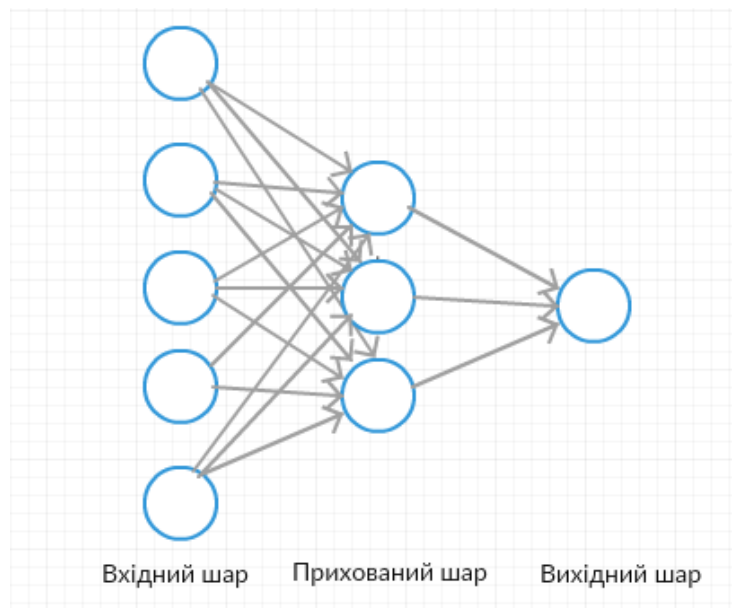


Рисунок 3.1 – Структура нейронної мережі

Навчання нейронної мережі здійснюється «із вчителем» з використанням алгоритму зворотнього поширення помилки, який широко використовується для навчання багатошарових нейронних мереж з неперервними функціями активації [14].

В якості функції активації найчастіше використовується сигмоїдна функція, графік якої зображено на рисунку 3.2.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3.7)$$

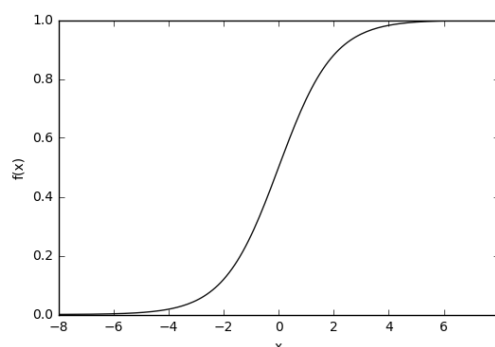


Рисунок 3.2 – Сигмоїдна функція активації

Перш ніж застосовувати нейронну мережу [15], необхідно здійснити попередню підготовку даних – часового ряду значень показника. Для цього використаємо метод ковзного середнього з величиною інтервалу



згладжування 3 для згладжування раптових коливань та підкреслення довготривалих трендів.

Для формування навчальної вибірки використаємо метод «часових вікон» [16]. Вхідне вікно формує дані для входів нейронної мережі, а вихідне – для виходів. Відповідна пара вхідного та вихідного векторів створюють одну реалізацію часового ряду. При зсуві часових вікон за часовим рядом з кроком  $s$ , отримаємо другу і наступні реалізації часового ряду. Задамо ширину вхідного вікна  $m$  рівною 5, вихідного шару  $p$  – рівною 1, крок  $s$  рівним 1.

Розділимо часовий ряд  $y$  співвідношенні 70:30 – 70% вибірки для навчання і 30% вибірки для тестування відповідно.

Отже, на вхід нейронної мережі подаються 5 значень показника  $Y_t, Y_{t+1}, Y_{t+2}, Y_{t+3}, Y_{t+4}$ , а на виході отримуємо прогнозоване значення показника на наступний момент часу  $Y_{t+5}$ ,  $t = 1, 2, \dots, T$ .

### 3.4.3 Оцінка точності прогнозів

Для оцінки точності прогнозів обчислюється середньоквадратичне відхилення за формулою:

$$S = \sqrt{\frac{\sum_{i=1}^n (Y'_i - Y_i)^2}{n}}, \quad (3.8)$$

де  $Y'_i$  – фактичні значення часового ряду;

$Y_i$  – обчислені значення часового ряду;

$n$  – число рівнів часового ряду.

### 3.5 Експериментальні дослідження

Реалізувавши кожен з наведених вище методів, провели експериментальні дослідження і отримали наступні результати.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

Використовуючи метод ковзного середнього для прогнозування, отримали результати, зображені на рисунку 3.3. Даний метод спрогнозував значення показника амоній іони на наступний рік рівним 0,35. У порівнянні із фактичними значеннями отримали значення середньоквадратичного відхилення рівне 0,19.



Рисунок 3.3 – Результати прогнозування методом ковзного середнього

Використовуючи метод експоненціального згладжування для прогнозування, отримали результати, зображені на рисунку 3.4. Даний метод спрогнозував значення показника амоній іони на наступний рік рівним 0,33. У порівнянні із фактичними значеннями отримали значення середньоквадратичного відхилення рівне 0,18.

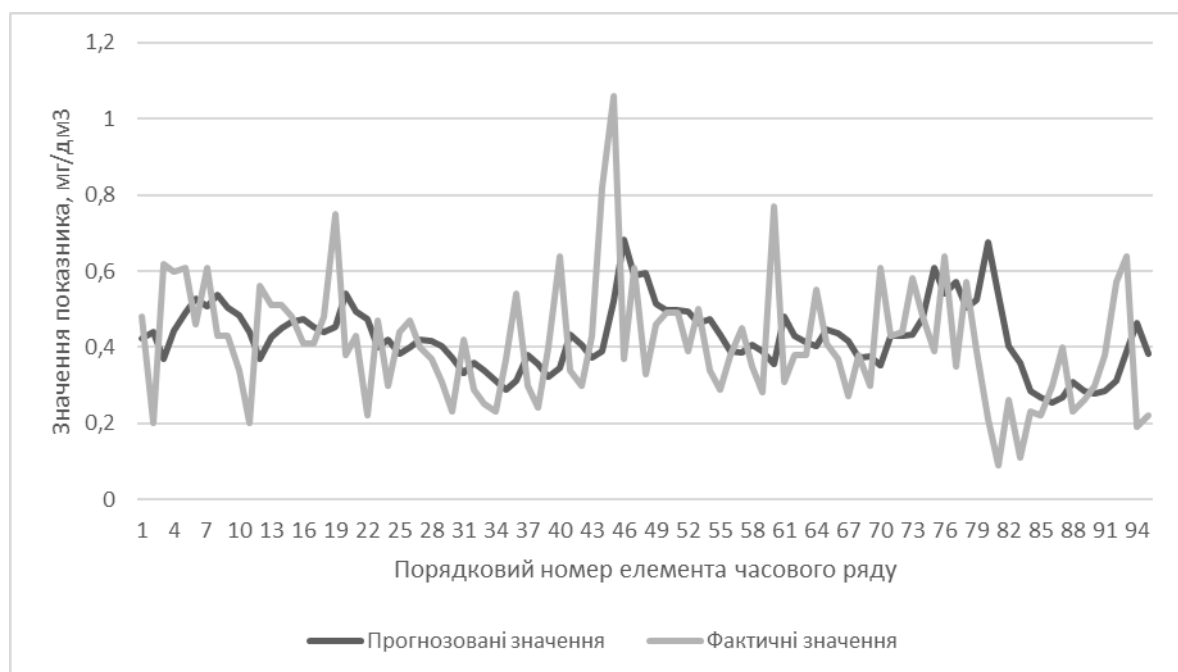


Рисунок 3.4 – Результати прогнозування методом експоненціального згладжування

Використовуючи метод найменших квадратів для прогнозування, отримали результати, зображені на рисунку 3.5. Даний метод спрогнозував значення показника амоній іони на наступний рік рівним 0,38. У порівнянні із фактичними значеннями отримали значення середньоквадратичного відхилення рівне 0,18.



Рисунок 3.5 – Результати прогнозування методом найменших квадратів

Використовуючи нейронну мережу для прогнозування, отримали результати, зображені на рисунку 3.6. Даний метод спрогнозував значення показника амоній іони на наступний рік рівним 0,33. У порівнянні із фактичними значеннями отримали значення середньоквадратичного відхилення рівне 0,09.



Рисунок 3.6 – Результати прогнозування з використанням нейронної мережі

У таблиці 3.1 наведено порівняння точності прогнозування за допомогою екстраполяційних методів та нейронних мереж.

Таблиця 3.1 – Порівняння точності методів прогнозування

Метод	Метод ковзного середнього	Метод експоненціального згладжування	Метод найменших квадратів	Нейронна мережа
Похибка	0,19	0,18	0,18	0,09
Прогноз	0,35	0,33	0,38	0,33

Отже, найкращу точність прогнозування отримали при використанні нейронної мережі – середньоквадратичне відхилення сягає лише 0,09. Методи екстраполяції прогнозують з приблизно однаковим

середньоквадратичним відхиленням - 0,18, що в 2 рази більше, ніж при використанні нейронної мережі. Проте, якщо порівняти спрогнозовані значення показника на наступний період, то видно, що всі методи видають приблизно схожий результат.

Отже, нейронна мережа прогнозує з найвищою точністю серед усіх використаних методів.

### Висновок до розділу

У даному розділі було сформульовано змістовну постановку задачі, яку необхідно розв'язати, та відповідну їй математичну постановку задачі. Розглядається задача прогнозування часових рядів для проведення ефективної екологічної оцінки водних ресурсів України. Обґрунтовано вибір методів розв'язання поставленої задачі та із всього різноманіття для проведення досліджень вибрано методи апроксимації та метод прогнозування з використанням нейронної мережі. Описано метод ковзного середнього, метод експоненціального згладжування, метод найменших квадратів та метод прогнозування за допомогою нейронної мережі. Нейронна мережа реалізована у вигляді тришарового персептрона. Було реалізовано кожен із вищенаведених методів і проведено експериментальні дослідження для визначення методу, що дає найкращі результати. В якості критерію використовується середньоквадратичне відхилення. Отримано результати, що підтверджують ефективність прогнозування з використанням нейронної мережі, точність результатів якої перевищує точність екстраполяційних методів у 2 рази.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

## 4 ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

### 4.1 Засоби розробки

Даний програмний продукт реалізовано на платформі Microsoft .NET у середовищі програмування Visual Studio 2015 з використанням фреймворку ASP.NET MVC, що реалізує шаблон Model-view-controller. При розробці системи використовується сервер баз даних Microsoft SQL Server (Microsoft SQL Server 2014 Express (x64) або Microsoft SQL Server 2014 Standart (x64)), який містить широкі можливості для розробки бізнес-логіки та подальшого розвитку системи [17].

У проекті використовуються геоінформаційні технології (ГІС), що реалізуються і підтримуються за допомогою сервісу від Google – Google Maps Api [18]. Google Maps Api – це інтерфейс, що дозволяє вбудовувати карти на зовнішні сайти за допомогою JavaScript.

.NET Framework – це програмна платформа, що була випущена компанією Microsoft у 2002 році. Основою платформи є середовище виконання Common Language Runtime (CLR), що підходить для різних мов програмування. Функціональні можливості CLR доступні в будь-яких мовах програмування, що використовують це середовище. Основною мовою програмування вважається C#.

.NET Framework – одна з найпопулярніших та найефективніших середовищ розробки. Ця повнофункціональна платформа підтримує розробку сучасних програмних продуктів і володіє високою сумісністю з існуючими застосуваннями та бібліотеками. Дане середовище має ряд переваг.

Це повна підтримка ООП. Бібліотека .NET є бібліотекою класів, а не функцій, і це означає, що можна створювати об'єкти цих класів, а також створювати нові класи, використовуючи бібліотечні в якості базових.

Це мовна незалежність. У середовищі .NET всі програми, не залежно від того, якою мовою вони написані, компілюються в певну проміжну мову

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Microsoft Intermediate Language (MSIL). Цим самим досягається мовна сумісність на рівні MSIL.

Це безпека коду. Середовище .NET забезпечує безпеку на рівні коду, строгий контроль типів проміжної мови дозволяє середовищу .NET перед запуском визначити політики безпеки.

Середовище .NET має вбудовану підтримку і створення web-служб, а також вбудовану підтримку динамічних web-сторінок [19].

Для розробки використовується ASP.NET MVC Framework, що реалізує MVC паттерн (рисунок 4.1). MVC шаблон отримав велику популярність в якості паттерна для веб-застосувань в силу 2 причин:

- необхідність веб-застосування об'єднувати декілька технологій;
- взаємодія користувача з MVC застосуванням відповідає наступному циклу: користувач робить дію, застосування у відповідь змінює свою модель даних і представляє користувачу оновлений вид, це дуже зручно для веб застосувань, що представляються у вигляді серії HTTP запитів і відповідей.



Рисунок 4.1 – MVC шаблон

ASP.NET MVC Framework реалізує MVC паттерн і, тим самим, забезпечує покращене розділення концепцій [20]. Основними перевагами даного фреймворку є розширюваність, повний контроль над HTML і HTTP, простота супроводу і тестування, могутня система маршрутизації, сучасний API і відкритість вихідного коду.

Також в розробці використовуємо Microsoft SQL Server - комерційна система керування базами даних, що розповсюджується корпорацією Microsoft. Мова, що використовується для запитів — Transact-SQL. Використовується як для невеликих і середніх за розміром баз даних, так і для великих баз даних масштабу підприємства. Багато років вдало конкурує з іншими системами керування базами даних [21]. Основними перевагами є:

- масштабованість;
- висока продуктивність;
- простота використання;
- інтеграція з іншими продуктами Microsoft (Access, Excel).

## 4.2 Вимоги до технічного забезпечення

### 4.2.1 Загальні вимоги

Для нормального функціонування системи, необхідний наступний комплекс технічних засобів:

- операційна система для центральних серверів баз даних і додатків Microsoft Windows Server 2014 R2 Standard (x64);
- система управління базами даних - Microsoft SQL Server 2014 Express (x64) або Microsoft SQL Server 2014 Standart (x64);
- веб-сервер: IIS v.8.5 або вище;
- обов'язкове системне програмне забезпечення - Framework v. 4 / 4.5.2; ASP.NET MVC 3,4; WebDeploy 3.5 (x64);



- програмний генератор звітів DevExpress ver.15;
- програмний сервіс Google Map;
- клієнтське програмне забезпечення на базі Microsoft Office 2007 і вище;
- серверний антивірус;
- операційна система клієнтських робочих станцій – Microsoft Windows 7, 8 і вище;
- браузері клієнтських робочих станцій - Microsoft Internet Explorer версії 9.0 або вище, Google Chrome, Mozilla Firefox;
- комплект антивірусів для робочих станцій.

#### 4.3 Архітектура програмного забезпечення

Для розробки даної системи було обрано багатошарову архітектуру. Як правило, виділяються шари користувацького інтерфейсу, бізнес-логіки та шар доступу до даних. В рамках такої архітектури користувачі виконують запити через шар користувацького інтерфейсу. Цей шар взаємодіє лише із шаром бізнес-логіки. Шар бізнес-логіки може викликати шар доступу до даних для обробки запитів користувача. Таким чином, для кожного шару чітко визначені його обов'язки.

Для роботи системи використовуються 2 окремі бази даних – Admin\_DB для збереження службових даних про користувачів системи та EcoWaterMon\_DB для збереження всіх необхідних даних моніторингу та водогосподарської обстановки. Схема структурна архітектури програмного забезпечення зображена на рисунку 4.2.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

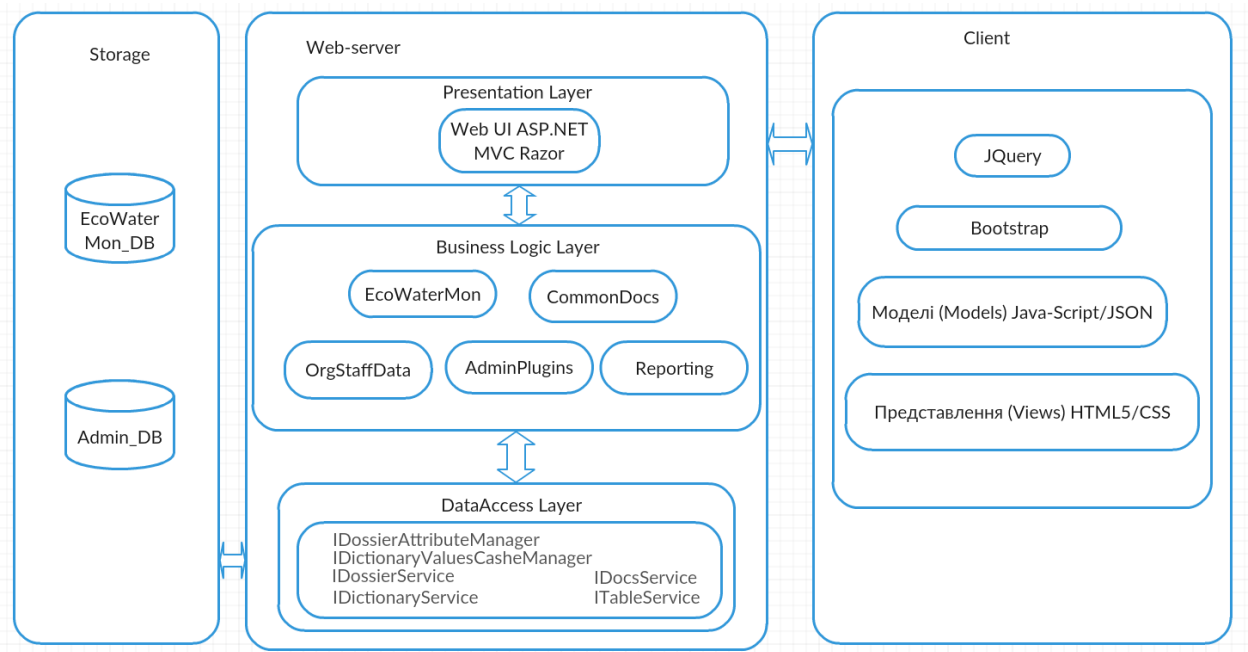


Рисунок 4.2 – Схема структурна архітектури програмного забезпечення

#### 4.3.1 Діаграма класів

Схема структурна класів системи наведена в частині графічного матеріалу.

Нижче описано призначення основних сервісів моніторингу та ведення водогосподарської обстановки.

Інтерфейс `IGDKMapService` – сервіс, що відповідає за виведення інформації на останню дату спостереження у порівнянні з ГДК.

Інтерфейс `IMapEcoWaterMonService` – сервіс, що відповідає за відображення постів спостереження на карті та їх фільтрацію.

Інтерфейс `IParsingDocFileService` – сервіс, що відповідає за загрузку даних з документу.

Інтерфейс `IPostsService` – сервіс, що відповідає за відображення дерева показників для моніторингу при реєстрації пункту спостереження.

Інтерфейс `IWaterResourcesService` – сервіс, що відповідає за ведення моніторингу рівня зарудненості водних об'єктів.

Інтерфейс `IWaterSituationService` – сервіс, що відповідає за ведення водогосподарського стану.

### 4.3.2 Діаграма послідовності

Схема структурна послідовності для функції перегляду карти перевищення ГДК показників на постах спостереження та можливості фільтрації об'єктів на карті наведена в частині графічного матеріалу.

Схема структурна послідовності, що відображає процес додавання даних моніторингу стану водних об'єктів наведена в частині графічного матеріалу.

### 4.3.3 Діаграма компонентів

Схема структурна компонентів представлена в частині графічного матеріалу. WaterMonitoring – компонент для ведення моніторингу забруднення водних ресурсів. WaterSituation – компонент для контролю водогосподарської обстановки. Admin – компонент для адміністрування системи. DataAccess – компонент доступу до даних. CommonDocs – компонент базових операцій з картками. Reporting – компонент для формування звітності. GoogleMaps – компонент для Google Maps API.

### 4.3.1 Специфікація функцій

У таблицях 4.1 – 4.5 наведено опис основних функцій програмних модулів.

Таблиця 4.1 – Опис функцій EcoWaterMon.Plugin

Назва	Опис
Клас GDKMapService	
public List<LastGDKData> GetLastData()	Метод, що повертає значення показників моніторингу по постах спостереження на останню дату спостережень
Інтерфейс IMapEcoWaterMonService	

## Продовження таблиці 4.1

Назва	Опис
public IEnumerable<ListModel> GetDicList()	Метод для отримання значення словника (для фільтру)
public List<EcoWaterMonArea> FilterMap(decimal?[] RiverBasDec, decimal?[] LaboratoryDec, decimal?[] WaterCompNameDec, decimal?[] WaterDepNameDec, decimal?[] OPTYPEDec, decimal?[] PeriodOfContDec, decimal?[] MonTypeDec, decimal?[] NormTypeDec, decimal?[] TransOPDec)	Метод для фільтрація постів спостереження на карті залежно від вибраних параметрів
Інтерфейс IParsingDocFileService	
public string ParseDoc(Stream fileStream, decimal dossierId)	Метод парсингу документу
string FillGrid(ParagraphCollection pc, decimal dossierId);	Метод для заповнення таблицки отриманими даними із документа
decimal GetDicIdByDicCode(string code, decimal dicId);	Метод для отримання ідентифікатору значення словника по його локальному коду
ModelForGrid SelectData(ParagraphCollection pc, string firstLine, string secondLine, int n);	Метод для вибірки даних із документу для створення одного запису в ґриді
string GetDate(decimal dossierId);	Метод, що повертає дату проведення моніторингу

## Продовження таблиці 4.1

Назва	Опис
public string DeleteRecords(decimal dossierId, decimal sectionId, decimal docCategory, bool saveRowDB)	Метод для видалення із таблиці даних про водосховища
Інтерфейс IPostsService	
public void InsertGridValue(List<TreeModel> arr, decimal dossierId, decimal sectionId)	Метод для запису назв вибраних показників в грід
public List<TreeModel> GetGridData(decimal dossierId, decimal sectionId)	Метод для отримання назв показників із гріда
Інтерфейс IWaterResourcesService	
public Dossier29003Model ChangeGridAndValues(decimal attr29034, bool editType)	Метод для зміни гріда з показниками для моніторингу при зміні вибраних показників
public IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad)	Метод для отримання списку даних моніторингу забруднення водойм
public void DeleteRow(decimal dossierId, decimal sectionId, string date)	Метод для видалення запису моніторингу забруднення водойм
public List<string> FillDateArray(decimal dossierId, decimal sectionId)	Метод для отримання всіх дат моніторингу по даному посту спостереження
Інтерфейс IWaterSituationService	
decimal GetMonth(decimal dictionaryId, string localCode);	Метод, що повертає місяць по локальному коду

## Продовження таблиці 4.1

Назва	Опис
IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad, decimal dicId);	Метод для отримання даних для відображення в ґриді
string EditRow(decimal dossierId, decimal sectionId, decimal dicId, bool onlyEdit, string jsonObjectString);	Метод для додавання/редагування запису в ґриді

Таблиця 4.2 – Опис функцій Reporting.Plugin

Назва	Опис
Інтерфейс IReportingManager	
IEnumerable<JsTreeJsonModel> GetReportTree();	Метод, що повертає кеш дерева звітів. Якщо відсутній – створення кешу і повернення
IEnumerable<Report> GetReports()	Метод, що повертає список всіх звітів і кешує їх
void ClearCache()	Метод для очистки кешу
Інтерфейс IReportingService	
IQueryable<Report> GetReportList(string code, string name);	Метод, що повертає список звітів за кодом і назвою
bool CheckReportNumber(string number, int? id)	Метод, що визначає чи існує звіт із заданим кодом
IEnumerable<SelectListItem> GetOperationalParamTypes()	Метод, що повертає список типів параметрів для формування звітів

## Продовження таблиці 4.2

Назва	Опис
Report CreateOperativeXML(Report data, List<OperationalReportParam> parameters)	Метод для створення XML документу з параметрами для формування звіту
Tuple<Report, List<OperationalReportParam>> ParseOperativeXML(Report report)	Метод для парсингу XML документу з параметрами
IEnumerable<JsTreeJsonModel> GetNodes(IEnumerable<Report> reports)	Метод, що повертає вершини для побудови дерева звітів
IEnumerable<Report> GetAllReports()	Метод, що повертає список звітів
ViewReport GetReportById(int id)	Метод, що повертає звіт по ід
bool ValidateQuery(string query, out string errorMessage)	Метод для перевірки коректності запиту для формування звіту
bool IsDisplay(Report report, object data)	Метод, що перевіряє чи можна відображати даний звіт на формі
Інтерфейс IUserReportManager	
IUserReport GetNewReport()	Метод, що повертає нову обгортку звіту
IUserReport GetReportById(int id)	Метод, що повертає обгортку існуючого звіту за ідентифікатором
IUserReport GetReportByCode(string code)	Метод, що повертає обгортку існуючого звіту за кодом
Інтерфейс IUserReportService	
DataSet CreateEmptyDataSet(string query)	Метод для створення пустого DataSet

Продовження таблиці 4.2

Назва	Опис
DataSet CreateFilledDataSet(string query, Dictionary<string, string> data)	Метод для створення заповненого DataSet
void InitReport(int id, IUserReport report)	Метод для ініціалізації шаблону звіту по ідентифікатору
void InitReport(string code, IUserReport report)	Метод для ініціалізації шаблону звіту по коду
Report GetReportById(int id)	Метод, що повертає звіт по ідентифікатору
Інтерфейс IUserReport	
void UpdateDataSource(string query)	Метод для оновлення DataSource звіту за результатом запиту
void PrepareReport(Dictionary<string, string> data, Dictionary<string, string> reportParams)	Метод для підготовки звіту – заповнення DataSource та параметрів
void LoadTemplate(Stream stream)	Метод для завантаження шаблону звіту з потоку
void SaveTemplate(Stream stream)	Метод для збереження шаблону в потік

Таблиця 4.3 – Опис функцій OrgStaffData

Назва	Опис
Інтерфейс IOrgStaffTreeCasheManager	
List<JsTreeJsonModel> GetJsTree(int clientId)	Метод, що повертає дерево активних елементів ОШС у вигляді JsTree



## Продовження таблиці 4.3

Назва	Опис
List<JsTreeJsonModel> GetJsDepartmentsTree(int clientId)	Метод, що повертає дерево активних підрозділів і клієнтів для батьківського елемента
OrgStaffNodeCollection GetNodesTree()	Метод, що повертає дерево активних елементів ОШС у вигляді колекції вузлів
void ClearCache()	Метод для очистки кешу
Інтерфейс IOrgStaffTreeService	
List<JsTreeJsonModel> GetNodes(decimal? nodeId, List<Tree_Staff> cacheTree, bool withEmployeesCounts = true)	Метод, що повертає всі елементи дерева із cacheTree, що є дочірніми для елемента node
JsTreeJsonModel GetSubNodes(decimal parentId, List<Tree_Staff> cacheTree, bool withEmployeesCounts = true)	Метод, що повертає один елемент в першому рівні та всі його підлеглі елементи в подальших рівнях
List<Tree_Staff> GetActiveTree(params TreeRecordTypes[] types)	Метод, що повертає всі активні елементи Tree_Staff
List<JsTreeJsonModel> GetDepartmentsWithDocsCounts(decimal docCategory)	Метод, що повертає всі активні структурні підрозділи з таблиці Tree_Staff, що прив'язані до певної категорії документів у вигляді списку із відображенням кількості документів цієї категорії, прив'язаних до кожного з підрозділів

## Продовження таблиці 4.3

Назва	Опис
OrgStaffNodeCollection GetOrgStaffTree(List<Tree_Staff> cacheTree)	Метод, що повертає дерево у вигляді вкладених колекцій OrgStaffNodeCollection

Таблиця 4.4 – Опис функцій DataAccess

Назва	Опис
Інтерфейс IDossierAttributeManager	
DossierAttribute LoadAttribute(decimal dossierId, decimal sectionId, decimal gridAttrId, bool refresh = false, decimal docCategory = 0)	Метод для отримання атрибутів досьє
DossierAttribute LoadGridAttribute(decimal dossierId, decimal sectionId, decimal gridAttrId, bool refresh = false, decimal docCategory = 0)	Метод для отримання атрибутів ґріда
DossierAttribute GetGridItem(decimal dossierId, decimal sectionId, decimal gridAttrId, decimal? itemId, decimal docCategory = 0)	Метод для отримання елемента ґріда
List<GridDossierAttribute> LoadAttributes(decimal dossierId, decimal sectionId)	Метод для отримання атрибутів ґридів досьє
bool DeleteGridAttribute(decimal dossierId, decimal sectionId, decimal gridAttrId)	Метод для видалення атрибутів ґріда для досьє

## Продовження таблиці 4.4

Назва	Опис
bool DeleteGridAttributeAll(decimal dossierId)	Метод для видалення всіх атрибутів ґридів для досьє
Інтерфейс IDictionaryValuesCacheManager	
List<DictionaryValue> GetValues(params decimal[] dictionaryIds)	Метод для отримання значень вибраних словників
DictionaryValue GetValue(decimal dictionaryId, decimal valueId)	Метод для отримання вибраного значення із вибраного словника
DictionaryAttribute GetDictionaryAttribute(decimal dictionaryId)	Метод для отримання атрибутів словника
List<DictionaryData> GetDictionaries()	Метод для отримання списку всіх довідників
DictionaryData GetDictionary(decimal dictionaryId)	Метод для отримання довідника з атрибутами і значеннями
DictionaryData GetDictionaryByAttributeId(decimal dictionaryAttributeId)	Метод для отримання всього довідника з атрибутами і значеннями по ідентифікатору атрибуту
void Insert(DictionaryValue dictionaryValue)	Метод для додавання нового значення в словник
Void ClearCacheDictionary(decimal dictionaryId)	Метод для очистки кешу словника
Інтерфейс IDossierService	
DossierData CreateDossier(decimal id)	Метод для створення порожнього досьє

## Продовження таблиці 4.4

Назва	Опис
DossierData CreateDossierBySectionId(decimal id)	Метод для створення порожнього досьє по ідентифікатору секції
DossierData CreateDossierByAttributeId(decimal attributeId)	Метод для створення порожнього досьє по ідентифікатору атрибута
DossierData LoadDossier(decimal itemId)	Метод для завантаження екземпляру досьє з бази даних
DossierData LoadDossierBySectionId(decimal sectionItemId)	Метод для завантаження екземпляру досьє з бази даних по ідентифікатору секції
decimal GetTemplateIdByItemId(decimal itemId)	Метод, що повертає ідентифікатор шаблону досьє по ідентифікатору екземпляру
decimal GetTemplateIdBySectionItemId(decimal sectionItemId)	Метод, що повертає ідентифікатор шаблону досьє по ідентифікатору екземпляру секції
void SaveDossier(DossierData dossier, decimal[] sectionIds)	Метод для збереження екземпляру досьє в базу даних
void DeleteDossier(decimal itemId)	Метод для видалення екземпляру досьє
IQueryable GetDossierList(decimal sectionId, string selectTemplate, decimal? departmentId = null)	Метод для отримання переліку значень заданої секції досьє
Інтерфейс IDictionaryService	
List<DictionaryData> LoadDictionaryList()	Метод для завантаження переліку довідників

## Продовження таблиці 4.4

Назва	Опис
DictionaryData LoadDictionary(decimal dictionaryId, bool loadValues = true)	Метод для завантаження довідника з атрибутами та значеннями
DictionaryData LoadDictionaryByAttributeId(decimal dictionaryAttributeId, bool loadValues = true)	Метод для завантаження довідника по ідентифікатору атрибуту
DictionaryAttribute LoadDictionaryAttribute(decimal dictionaryId, bool loadValues = true)	Метод для завантаження атрибутів довідника із значеннями
List<DictionaryAttribute> LoadDictionaryAttributes(decimal[] dictionaryIds, bool loadValues = true)	Метод для завантаження атрибутів декількох довідників із значеннями
void Insert(DictionaryValue dictionaryValue)	Метод для додавання значення в довідник

Таблиця 4.5 – Опис функцій CommonDocs

Назва	Опис
Інтерфейс IDocsService	
DossierData CreateDoc(decimal dossierId, bool copy = false)	Метод для отримання шаблону документа
DossierData LoadDoc(decimal itemId, bool dbItem = false)	Метод для отримання екземпляру документа по ідентифікатору
DossierData LoadDocBySectionId(decimal itemId, bool refresh=false)	Метод для отримання екземпляру секції документа по ідентифікатору

Продовження таблиці 4.5

Назва	Опис
ResultOrError<DossierData> InsertDoc(DossierSection dossierSection, decimal? deptId)	Метод для додавання нової секції (вкладки) документа
ResultOrError<DossierData> UpdateDoc(decimal itemId, DossierSection dossierSection)	Метод для редагування нової секції (вкладки) документа
string DeleteDoc(decimal id, decimal itemId)	Метод для видалення документа
bool DossierIsDeleted(decimal dossierId)	Метод для перевірки чи видалено досьє (документ)
IQueryable GetDossierList(decimal docCategory, string selectTemplate, decimal? departmentId = null)	Метод для отримання списку документів вибраної категорії, прив'язаних до конкретного підрозділу
string GetPageTitle(decimal id, decimal itemId)	Метод для отримання заголовку сторінки
decimal[] GetVisiblePages(decimal id, decimal itemId = 0, Feature featureToCheck = Feature.ViewSection, DossierData dosData = null)	Метод для отримання переліку доступних користувачу розділів досьє
decimal GetDocumentDepId(decimal itemId)	Метод для отримання ідентифікатору департменту, в якому знаходиться документ
UserInfo GetUserInfo()	Метод для отримання інформації про поточного користувача

## Продовження таблиці 4.5

Назва	Опис
List<Dossier_Button> GetDossierButton(decimal dossierId)	Метод для отримання списку кнопок для досьє
Інтерфейс ITableService	
DossierAttribute LoadGridAttribute(decimal dossierId, decimal sectionId, decimal gridAttrId, bool refresh = false, decimal docCategory = 0)	Метод для отримання атрибутів ґріда
DossierAttribute GetGridItem(decimal dossierId, decimal sectionId, decimal gridAttrId, decimal? itemId, decimal docCategory = 0);	Метод для отримання елемента ґріда
ResultOrError<DossierData> EditGridItem(decimal dossierId, decimal sectionId, DossierAttribute dosAttribute, HttpPostedFileBase fileSource, decimal docCategory = 0, decimal saveRowDB = 0)	Метод для редагування/додавання елемента ґріда
Void AddGridItem(DossierAttribute newAttribute, DossierAttribute dosAttribute)	Метод для додавання (в пам'яті) елемента ґріда
void AddSectionItemFile(DossierSection sectionData, HttpPostedFileBase fileSource)	Метод для додавання атрибуту типу файл
string DeleteGridItem(decimal dossierId, decimal sectionId, decimal gridAttrId, decimal itemId, decimal docCategory = 0, bool saveRowDB = false, bool CascadingDelete = false)	Метод для видалення елемента ґріда
file GetGridItemFile(decimal dossierId, decimal sectionId, decimal gridAttrId, decimal gridAttrItemId, decimal gridFileAttrId)	Метод для завантаження файлу елемента ґріда

Продовження таблиці 4.5

Назва	Опис
file GetGridItemFile(decimal gridItemFileId)	Метод для завантаження файлу з бази даних по ідентифікатору запису з файлом.

#### 4.4 Опис звітів

На рисунках 4.6 – 4.10 наведено приклади сформованих звітів.

##### Кількість ПС по Головним річковим басейнам

18 травня 2019 р. 20:56

№ з/п	Назва Головного річкового басейну	Кількість (П)остів (С)постереження
1	Вісла	16
2	Дніпро	210
3	Дністер	56
4	Дон	42
5	Дунай	62
6	Південний Буг	42
7	Річки Приазов'я	3
8	Річки Причорномор'я	15

Рисунок 4.6 – Приклад звіту «Кількість ПС по Головним річковим басейнам»



## Вимірювальні станції

18 травня 2019 р. 20:59

№ з/п	Назва посту спостереження	Відомчий код ПС	Географічні координати	
			Широта	Довгота
1	Балка Ташлик, 6 км, Ташлицьке вдсх., став-охолоджувач ПУ АЕС (міст через Ташлицьке вдсх., південь м. Південно-Українськ)	ЧЕР Ю.БУГ 0150 0000 0000 0000 0000 0006 01 48	47° 48' 27,1"	31° 12' 20,7"
2	канал Дніпро-Донбас, 0,5 км, с. Шульгівка, після ГВК Дніпро-Донбас	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 01 12	48° 44' 20"	34° 18' 50"
3	канал Дніпро-Донбас, 109 км, м. Перещепино, міст, Перещепинський водозабір	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0109 02 12	49° 1' 47,6"	35° 22' 41"
4	канал Дніпро-Донбас, 170 км, с.Орілька, Орільське водосховище, шлюзовий попуск	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0170 03 12	48° 58' 48,6"	36° 2' 4,8"
5	канал Дніпро-Донбас, 181 км, р.Орілька, міст, 1,5 км від гирла,	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0181 04 12	49° 2' 4,6"	36° 6' 13,6"
6	канал Дніпро-Донбас, 214 км, м. Харків, Краснопавлівське водосховище, в/з міста	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0214 05 12	49° 8' 20,2"	36° 27' 46,9"
7	канал Дніпро-Донбас, 215 км, верхній б'єф Краснопавлівського водосховища, скид у р. Сіверський Дінець	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0215 06 12	49° 6' 49,8"	36° 30' 3,4"
8	канал Дніпро-Донбас, 215 км, верхній б'єф Краснопавлівського водосховища, скид у р. Сіверський Донець	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0215 06 12	49° 6' 46,9"	36° 30' 3,6"
9	канал Дніпро-Донбас, 216 км, нижній б'єф Краснопавлівського вдсх.	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0216 07 63	49° 6' 50,9"	36° 30' 28"
10	канал Дніпро-Донбас, 256 км, с. Грушеваха	ЧЕР ДНЕПР 0490 0000 0000 0000 0000 0000 0256 08 63	49° 7' 32,9"	36° 54' 12,6"
11	канал Дніпро-Кривий Ріг, 43 км, Південне в-ще, питний в/з	ЧЕР ДНЕПР 0196 0000 0000 0000 0000 0000 0043 01 12	47° 46' 53,8"	33° 33' 23,4"
12	Каховський магістральний канал, 127 км, с. Ганно-Муїївка, 1 км Приазовської ЗС	ЧЕР ДНЕПР 0106 0000 0000 0000 0000 0000 0127 01 23	46° 33' 32,4"	34° 51' 10,8"
13	оз. Кагул, 0 км, ГНС Нагірне	ЧЕР ДУНАЙ 0150 0000 0000 0000 0000 0000 0000 01 51	45° 23' 12,3"	28° 26' 26,1"
14	оз. Катлабух, 0 км, ГНС Кірова	ЧЕР ДУНАЙ 0150 0000 0000 0000 0000 0000 0000 02 51	45° 27' 43"	29° 3' 4,8"
15	оз. Катлабух, 0 км, Суворовська ГНС	ЧЕР ДУНАЙ 0150 0000 0000 0000 0000 0000 0000 01 51	45° 34' 38,5"	28° 57' 17"

Рисунок 4.7 – Приклад звіту «Пости спостереження»

**Моніторинг за даними поста спостереження (абсолютні значення)**

з 01.05.2000 р. до 01.05.2019 р.

По посту: р. Кізка, 5 км, с.Демидів, вплив зворотних вод "Агромарс"

Значення	Показники								
	Амоній-іони, мг/дм <sup>3</sup>	Біохімічне споживання кисню за 5 діб, мгО <sub>2</sub> /дм <sup>3</sup>	Завислі (суспендовані) речовини, мг/дм <sup>3</sup>	Кисень розчинений, мгО <sub>2</sub> /дм <sup>3</sup>	Нітрат-іони, мг/дм <sup>3</sup>	Нітрит-іони, мг/дм <sup>3</sup>	Сульфат-іони, мг/дм <sup>3</sup>	Фосфат-іони (поліфосфати), мг/дм <sup>3</sup>	Хлорид-іони, мг/дм <sup>3</sup>
Дата									
06.07.2004	25,26	95,60	123,80	1,17	0,25	0,02		11,40	60,20
01.09.2004	44,30	155,30	47,50	2,34	5,17	0,08	142,00	172,40	282,40
18.01.2005	5,02	6,60	4,20	3,60	0,70	0,06	57,00	1,21	26,20
01.02.2005	4,45	22,40	17,90	0,70	1,80	0,05	64,20	6,44	100,40
29.03.2005	6,40	26,00	8,70	7,10	9,80	2,07	67,80	0,55	67,30
05.04.2005	2,82	2,50	2,60	6,20	4,30	0,08	49,70	0,87	21,30
17.05.2005	4,81	4,20	16,10	1,80	0,50	0,05	49,80	1,95	21,30
21.06.2005	16,46	27,80	19,20	3,10	1,30	0,05	56,90	19,73	561,40
12.07.2005	26,45	7,60	20,00	7,90	0,70	0,01	35,30	12,52	45,10
11.08.2005	6,31	2,40	4,00	2,30	0,50	0,03	42,80	1,53	23,70
06.09.2005	32,03	136,20	15,80	1,33	1,78	0,01	84,70	35,76	123,80
18.10.2005	8,16	3,61	0,50	2,43	0,96	0,07	38,60	3,21	27,00
10.11.2005	7,00	5,70	15,20	2,40	0,30	0,04	31,40	5,00	27,00

Рисунок 4.8 – Приклад звіту «Загальні дані моніторингу. Моніторинг за даними поста спостереження»

Змн.	Арк.	№ докум.	Підпис	Дата

ІНФОРМАЦІЯ  
ПРО ВОДОГОСПОДАРСЬКУ ОБСТАНОВКУ В УКРАЇНІ  
на 1 лютого 2018 р.

№ з/п	Водоймище	Проектні дані		Фактичні дані				Вільний обсяг, млн.м3
		Рівень, м	Обсяг, млн.м3	Рівень, м	Обсяг, млн.м3	Приплив, м3/с	Скид, м3/с	
Р. ДНІПРО								
1	Київське	103,00	3730,00	102,29	3132,00	1230,00	1310,00	-598,00
2	Канівське	91,50	2620,00	91,40	2552,00	1600,00	1740,00	-68,00
3	Кременчуцьке	81,00	13520,00	78,80	9080,00	1780,00	2050,00	-4440,00
4	Кам'янське	64,00	2463,00	63,91	2417,00	2100,00	2060,00	-46,00
5	Дніпровське	51,40	3325,00	51,23	3261,00	2090,00	2090,00	-64,00
6	Каховське	16,00	18190,00	15,40	16905,00	0,00	2080,00	-1285,00
Разом:		43848,00		37347,00		(85,17 %)		-6501,00
Р. СІВЕРСЬКИЙ ДОНЕЦЬ								
7	Печенізьке	100,50	383,00	100,23	359,70	15,40	15,00	-23,30
8	Червоноокопське	72,50	477,00	71,87	377,20	28,00	39,00	-99,80
Разом:		860,00		736,90		(85,69 %)		-123,10
Р. ДНІСТЕР								
9	Дністровське	121,00	3000,00	118,90	2702,00	107,00	152,00	-298,00
Разом:		3000,00		2702,00		(90,07 %)		-298,00
Р. ПІВДЕННИЙ БУГ								
10	Ладизинське	177,00	150,00	-	-	-	-	-
Разом:		150,00		0,00		0,00		

Рисунок 4.9 – Приклад звіту «Інформація про водогосподарську обстановку в Україні»

ВОДОГОСПОДАРСЬКІ СИСТЕМИ

№ з/п	Найменування каналів, зрошувальних систем	Забор води з початку сезону, тис.м3	Декадний план, тис.м3	Забор води з початку декади, тис.м3	Середньодобовий забор води, м3/сек
1	Дніпро-Донбас	83462,40	1471,00	3628,80	10,50
2	Ігулецька	0,00	9509,00	0,00	0,00

ГІДРОХІМІЧНА ОБСТАНОВКА

№ з/п	Річка-Пункт	Загальна мінералізація, мг/л		Хлориди, мг/л		Жорсткість, мг.екв/л	
		Норма	Факт	Норма	Факт	Норма	Факт
1	Ігулець - Ігулецька ГНС	-	-	350,00	460,00	-	-

Довідки по телефону 235 - 48 - 99 (Відділ водних відносин)  
579 - 21 - 15 (диспетчерський)

Рисунок 4.10 – Продовження прикладу звіту «Інформація про водогосподарську обстановку в Україні»

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		78

### Висновок до розділу

У даному розділі наведено опис програмного та технічного забезпечення. В якості засобів розробки обрано платформу Microsoft .NET, середовище програмування Visual Studio 2015. Система розробляється з використанням фреймворку ASP.NET MVC, що реалізує шаблон Model-view-controller. При розробці системи використовується сервер баз даних Microsoft SQL Server (Microsoft SQL Server 2014 Express (x64)). Визначено загальні вимоги до засобів технічного забезпечення для забезпечення повноцінної роботи системи.

Описано архітектуру програмного забезпечення. Для реалізації даної системи обрано багатoshарову архітектуру, в якій виділяється шар користувацького інтерфейсу, бізнес-логіки та шар доступу до даних.

На структурних схемах послідовності наведено опис одних із основних процесів системи – робота з картою перевищення ГДК показників на постах спостереження та додавання даних моніторингу водних об'єктів. Наведено структурні схеми класів системи та описано основні їхні методи. Описано структурну схему розгортання.

Інструментальні програмні засоби, що використовуються для розробки та експлуатації, забезпечують високий рівень кінцевого програмного продукту з використанням сучасних досягнень інформаційної та комп'ютерної технології.

## 5 ТЕХНОЛОГІЧНИЙ РОЗДІЛ

### 5.1 Керівництво користувача

Для того, щоб відкрити веб-систему «Моніторинг та екологічна оцінка водних ресурсів України», необхідно відкрити браузер і перейти за адресою <http://monitoring.davr.gov.ua>. Після цього відкриється головна сторінка системи, на якій відображається карта із зареєстрованими пунктами спостереження моніторингу забруднення водойм, що зображена на рисунку 5.1.



Рисунок 5.1 – Головна сторінка системи з картою перевищення ГДК показників

Кольоровими індикаторами позначається стан поста спостереження на останню дату досліджень. Розгорнувши блок «Інфо», що знаходиться в лівому верхньому кутку, можна побачити пояснення до кожного з індикаторів. Це зображено на рисунку 5.2.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		80





Рисунок 5.2 – Інформаційна довідка кольорів та розмірів індикаторів

Натискаємо на будь-який індикатор, що відображає пункт спостереження, і отримуємо віконечко з короткою інформацією про даний пост спостереження та таблицю із значеннями показників на останню дату спостереження у порівнянні із значеннями ГДК. Це зображено на рисунку 5.3.



Рисунок 5.3 – Коротка інформація про пост спостереження

Це віконечко містить посилання «Дані моніторингу», що дозволяє перейти на сторінку формування звіту, що зображена на рисунку 5.4.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Публічний користувач

Звітність / Перегляд звіту

Загальні дані моніторингу. Моніторинг за даними поста спостереження (10 показників).  
Таблично-графічне представлення даних

Період з: \*  
01.03.1990

Період до: \*  
31.03.2019

Показник: \*  
Всі

Норматив:  
Наказ МОЗ від 19.06.96 №

Виконати Очистити Закрити

© 2019, Інститут розробки інформаційних систем, Всі права захищено

Рисунок 5.4 – Введення параметрів для формування звіту

Ввівши необхідні параметри та натиснувши кнопку «Виконати», ми отримаємо звіт по загальним даним моніторингу у таблично-графічному вигляді, що зображено на рисунку 5.5 та рисунку 5.6.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		82

Державне агентство  
водних ресурсів України

## Моніторинг за даними поста спостереження (абсолютні значення)

з 01.03.1990 р. до 31.03.2019 р.

По посту: р. Дніпро, 854,5 км, водосховище, 500 м нижче БСА

Значення	Показники								
Дата	Амоній-іони, мг/дм3	Біохімічне споживання кисню за 5 діб, мгО2/дм3	Завислі (суспендовані) речовини, мг/дм3	Кисень розчинений, мгО2/дм3	Нітрат-іони, мг/дм3	Нітрит-іони, мг/дм3	Сульфат-іони, мг/дм3	Фосфат-іони (поліфосфати), мг/дм3	Хлорид-іони, мг/дм3
14.01.1993	1,70	2,20	3,90	8,90		0,65	39,00	1,00	38,00
19.01.1993	1,00	1,80	12,00	9,30	11,00	0,98	42,00	0,82	40,00
03.03.1993	1,80	2,90	5,30	7,80		0,16	44,00	0,57	30,00
23.03.1993	1,50	4,20	12,00	8,40	6,00	0,28	33,00		31,00
07.04.1993	1,70	3,20	7,80		8,00	0,46	42,00	0,82	32,00
05.05.1993	0,84	1,60	5,50	8,10	1,90	0,17	38,00	0,36	29,00
02.06.1993	1,32	2,20	13,30	8,10	2,60	0,20	45,00	0,77	27,00
07.07.1993	0,85	3,50	28,90	7,00	10,80	0,31	48,00	0,62	31,00
11.08.1993	0,46	2,70	9,40	7,60	3,60	0,13	20,00	0,35	28,00
30.09.1993	1,92	2,00	5,10	9,10	4,70	0,18	19,00	0,57	23,00
18.11.1993	1,60	1,90	1,20	10,50	4,80	0,09	23,00	0,38	27,00
08.12.1993	0,88	1,40	7,70		3,50	0,04	25,00	0,22	22,20
24.01.1994	2,00	1,80	19,00		6,70	0,38	27,00	0,60	34,00
02.03.1994	1,53	3,40	10,50		7,90	0,33	41,00	0,42	29,00
12.04.1994	0,84	1,30	8,60		3,60	0,04	21,00	0,13	12,00
21.04.1994	0,20	1,60	5,60		2,20	0,06	19,00	0,14	14,00
04.05.1994	0,37	1,50	5,70		0,60	0,02	21,00	0,17	10,00
08.06.1994	2,26	3,60	9,60		3,10	0,17	27,00	0,73	25,00

Рисунок 5.5 – Частина сформованого звіту у табличному вигляді

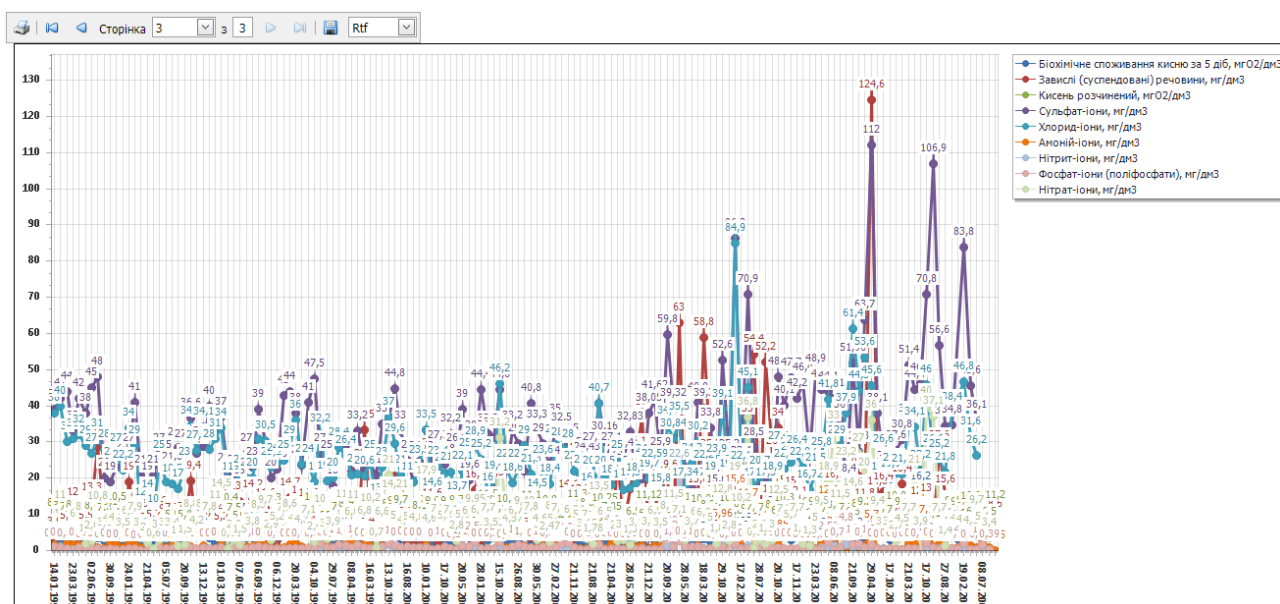


Рисунок 5.6 – Сформований звіт у графічному вигляді

Змн.	Арк.	№ докум.	Підпис	Дата

ДП ІС-5109.1181-с.ПЗ

Арк.

83



Натиснувши пункт меню «Карта», ми перейдемо на карту, що відображає всі зареєстровані пости спостереження. Карта зображена на рисунку 5.7.

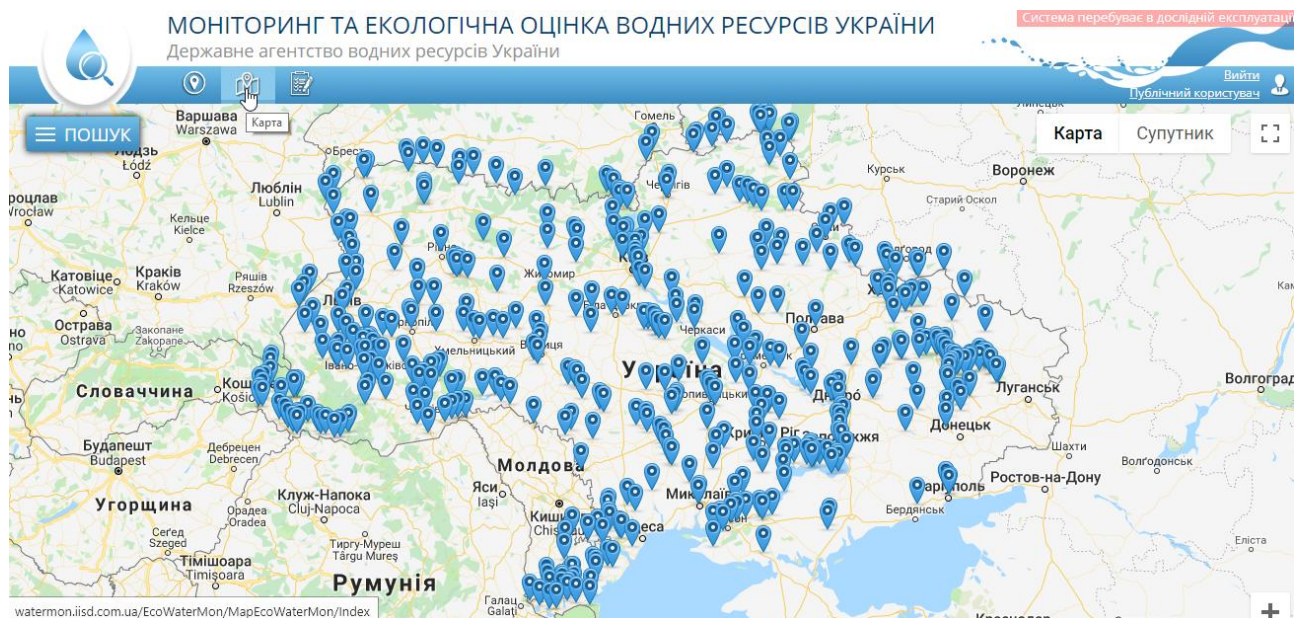


Рисунок 5.7 – Карта постів спостережень

Натиснувши на блок «Пошук», розкриється блок, що дозволяє відфільтрувати об'єкти карти за певними критеріями. Це зображено на рисунку 5.8.

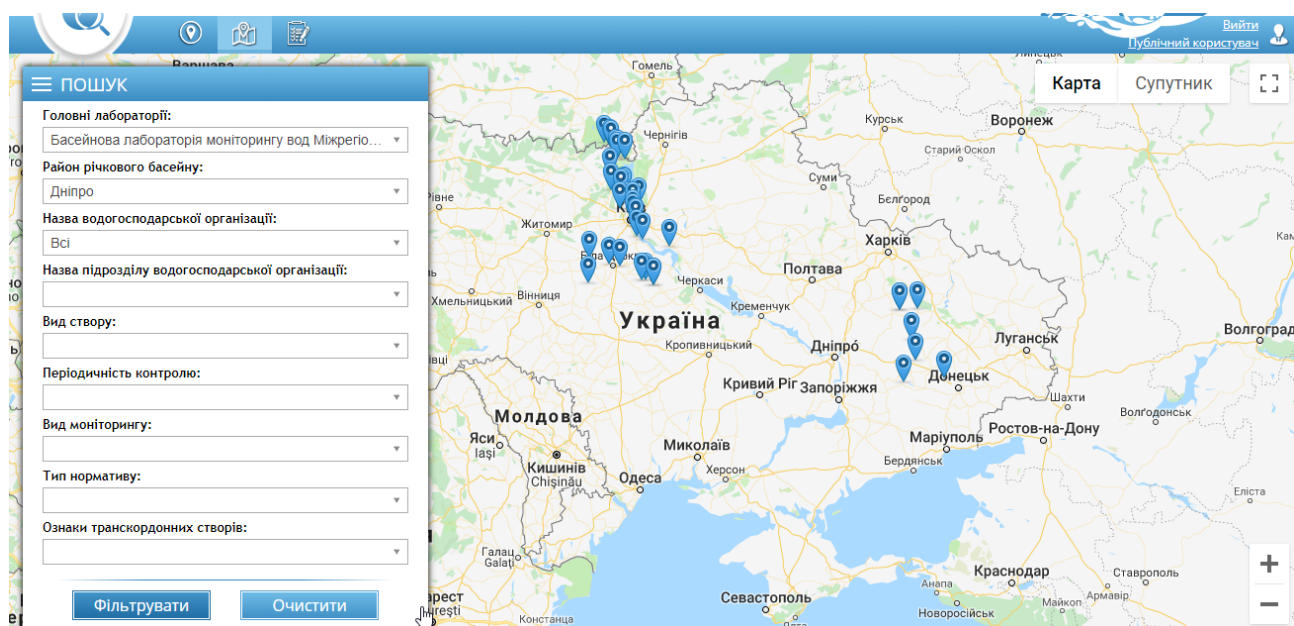


Рисунок 5.8 – Фільтрація постів спостереження

Змн.	Арк.	№ докум.	Підпис	Дата

Натиснувши на пункт меню «Звітність» незареєстрований користувач отримає доступ до звітів, що зображені на рисунку 5.9.

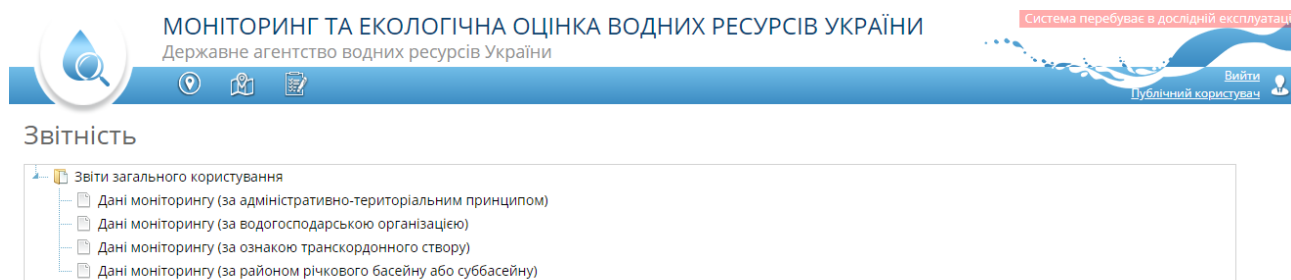


Рисунок 5.9 – Звітність

Для того, щоб увійти до системи, необхідно натиснути в правому верхньому куту кнопку «Вхід». На форму, що з'явиться, ввести логін і пароль. Це зображено на рисунку 5.10.

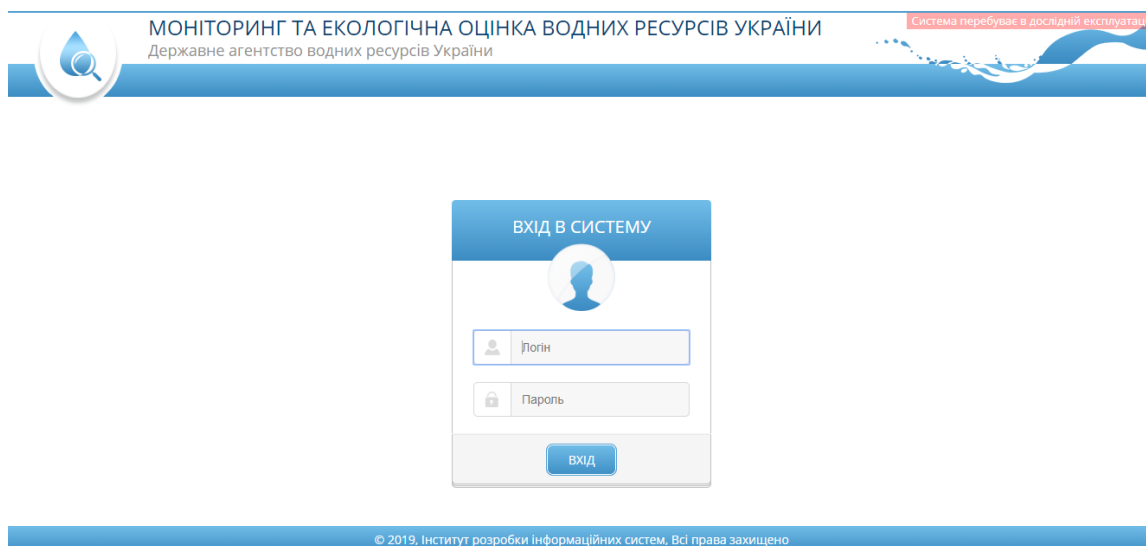


Рисунок 5.10 – Вхід в систему

Після входу в систему, обираємо пункт меню «Реєстр постів спостереження». На сторінці, що зображена на рисунку 5.11, натискаємо кнопку «Додати картку» для того, щоб зареєструвати новий пост спостереження.

## Реєстр постів спостереження

Фільтр по полям таблиці

Додати картку

1 з 5 100

Перегляд 1 - 100 з 467

	Повна назва ПС	Район річкового басейну	Водний об'єкт	Область	Відомчий код ПС	
<input type="checkbox"/>	руч. Знаменка, 5 км, смт. Білі Березки, нижче міста, кордон з РФ	Дніпро	руч. Знаменка	Сумська	ЧЕР ДНЕР 0892 0556 0013 0022 0000 0005 01 59	✗
<input type="checkbox"/>	р. Ягорлик, 20,2 км, с. Артирівка, кордон з Молдовою	Дністер	р. Ягорлик	Одеська	ЧЕР ДНЕСТР 0375 0000 0000 0000 0000 0020 01 51	✗
<input type="checkbox"/>	р. Ягодина, 2 км, нижче оз.Ягодинське	Вісла	р. Ягодина	Волинська	БАЛ ВИСЛА 0388 0510 0000 0000 0000 0002 06 07	✗
<input type="checkbox"/>	р. Шостка, 25 км, с. Гамалівка, 25 км, с.Гамалівка,вище техн.в/з м.Шостка	Дніпро	р. Шостка	Сумська	ЧЕР ДНЕР 0892 0500 0000 0000 0000 0025 01 59	✗
<input type="checkbox"/>	р. Шопурка, 1 км, смт В.Бичків, гирло	Дунай	р. Шопурка	Закарпатська	ЧЕР ДУНАЙ 1218 0927 0000 0000 0000 0001 01 21	✗
<input type="checkbox"/>	р. Шкло, 66 км, с. Краковець, кордон з Польщею, під мостом дороги м. Львів - с. Краковець	Вісла	р. Шкло	Львівська	БАЛ ВИСЛА 0665 0123 0000 0000 0000 0066 01 46	✗
<input type="checkbox"/>	р. Чорний Ташлик, 89 км, с. Звірівка, Новоукраїнського району	Південний Буг	р. Чорний Ташлик	Кіровоградська	ЧЕР Ю.БУГ 0196 0021 0000 0000 0000 0089 01 35	✗
<input type="checkbox"/>	р. Чорний Ташлик, 51 км, смт. Помічна, питний в/з селища	Південний Буг	р. Чорний Ташлик	Кіровоградська	ЧЕР Ю.БУГ 0196 0021 0000 0000 0000 0051 02 35	✗
<input type="checkbox"/>	р. Чечва, 16 км, смт. Рокитнятів, Чеченське вдсх.	Дністер	р. Чечва	Івано-Франківська	ЧЕР ДНЕСТР 1120 0039 0000 0000 0000 0016 01 26	✗
<input type="checkbox"/>	молдовою	Річки Причорномор'я	р. Чага	Одеська	ЧЕР КОГІЛЬ 0051 0000 0000 0000 0000 0066 01 51	✗

watermon.iisd.com.ua/CommonDocs/Docs/Add/29000

Рисунок 5.11 – Сторінка реєстру пунктів спостереження

Після цього відкриється сторінка для внесення даних по новому посту спостереження, що зображена на рисунку 5.12.

Реєстр постів спостереження

Зберегти Закрити

Опис місця розташування ПС/створу:  
нижче міста, кордон з РФ

Населений пункт, найближчий до створу:  
смт. Білі Березки

Район річкового басейну: \*  
Дніпро

Суббасейн:  
Десна

Назва головної лабораторії, що виконує ІЛВ:  
Лабораторія моніторингу вод та ґрунтів Регіонального офісу водних

Назва водогосподарської організації:  
РОВР у Сумській області

Організація, відповідальна за відбір проб:

Водний об'єкт

Водний об'єкт I категорії: \*  
р. Дніпро

Водний об'єкт II категорії:  
р. Десенка

Водний об'єкт III категорії:  
р. Знобовка

Водний об'єкт IV категорії:  
руч. Знаменка

Водний об'єкт V категорії:

Тип масиву поверхневих вод:

Код водогосподарської ділянки:

Відомчий код ПС:  
ЧЕР ДНЕР 0892 0556 0013 0022 0000 0005 01 59

Код ПС у Державному реєстрі:

Вид створу:

Тип нормативу:

Вид моніторингу:

Рисунок 5.12 – Реєстрація нового поста спостереження

Для даного поста спостереження обираються показники для

моніторингу забрудненості водних об'єктів, що зображені на рисунку 5.13.

## Показники

Групи показників	
<input type="checkbox"/> Гідробіологічні показники	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Фітопланктон, тис.клітин/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Біомаса, мг/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Хлорофіл, мг/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Мікробіологічні показники	<input checked="" type="checkbox"/>
<input type="checkbox"/> Показники радіаційної безпеки	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Цезій-137, пКі/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Стронцій-90, пКі/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Загальна об'ємна активність бета-випромінювачів, Бк/дм <sup>3</sup>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Ефективна питома активність природних радіонуклідів: калій-40, Бк/дм <sup>3</sup>	<input checked="" type="checkbox"/>

Рисунок 5.13 – Вибір показників для моніторингу

Вводимо географічні координати поста спостереження, натискаємо кнопку «Показати на карті» і даний пост спостереження відображається на карті. Чекбокс «Відображати на карті» є ознакою відображення даного поста спостереження на головній карті «Перевищення ГДК показників». Це зображено на рисунку 5.14.

Географічні координати

Десяткові	Градуси, минути, секунди
Широта: 52,355	Широта: 52° 21' 18"
Довгота: 33,5	Довгота: 33° 30' 0"

Відображати на карті: ☒

Системна інформація

Рисунок 5.14 – Продовження реєстрації поста спостереження





Для того, щоб додати новий показник, треба натиснути кнопку «Додати картку». Тоді відкриється сторінка додавання показника, що зображена на рисунку 5.17. Обираємо групу показника, показник.

Рисунок 5.17 – Додавання ГДК показника

Для додавання нового нормативу треба натиснути на кнопку додавання нового запису над таблицю «Нормативи». З'явиться віконечко, що зображено на рисунку 5.18, для вибору нормативу і введення максимального, мінімального допустимого значення показника.

Рисунок 5.18 – Додавання запису в таблицю «Нормативи»

Для ведення моніторингу забруднення водних об'єктів необхідно обрати пункт меню «Моніторинг водних ресурсів України». Відкриється перелік зареєстрованих постів спостереження, що зображено на рисунку 5.19.

Фільтр по полям таблиці

Стор. 1 з 5 100

Перегляд 1 - 100 з 465

	Повна назва ПС	Район річкового басейну	Водний об'єкт	Відомчий код ПС	
<input type="checkbox"/>	руч. Знаменка, 5 км, смт. Білі Березки, нижче міста, кордон з РФ	Дніпро	руч. Знаменка	ЧЕР ДНЕПР 0892 0556 0013 0022 0000 0005 01 59	✗
<input type="checkbox"/>	р. Ягорлик, 20,2 км, с. Артірівка, кордон з Молдовою	Дністер	р. Ягорлик	ЧЕР ДНЕСТР 0375 0000 0000 0000 0000 0020 01 51	✗
<input type="checkbox"/>	р. Ягодина, 2 км, нижче оз.Ягодинське	Вісла	р. Ягодина	БАЛ ВИСЛА 0388 0510 0000 0000 0000 0002 06 07	✗
<input type="checkbox"/>	р. Шостка, 25 км, с. Гамалівка, 25 км, с.Гамалівка,вище техн.в/з м.Шостка	Дніпро	р. Шостка	ЧЕР ДНЕПР 0892 0500 0000 0000 0000 0025 01 59	✗
<input type="checkbox"/>	р. Шопурка, 1 км, смт В.Бичків, гирло	Дунай	р. Шопурка	ЧЕР ДУНАЙ 1218 0927 0000 0000 0000 0001 01 21	✗
<input type="checkbox"/>	р. Шкло, 66 км, с. Краковець, кордон з Польщею, під мостом дороги м. Львів - с. Краковець	Вісла	р. Шкло	БАЛ ВИСЛА 0665 0123 0000 0000 0000 0066 01 46	✗
<input type="checkbox"/>	р. Чорний Ташлик, 89 км, с. Зеїрівка, Новоукраїнського району	Південний Буг	р. Чорний Ташлик	ЧЕР Ю.БУГ 0196 0021 0000 0000 0000 0089 01 35	✗
<input type="checkbox"/>	р. Чорний Ташлик, 51 км, смт. Помічна, питний в/з селища	Південний Буг	р. Чорний Ташлик	ЧЕР Ю.БУГ 0196 0021 0000 0000 0000 0051 02 35	✗
<input type="checkbox"/>	р. Чечва, 16 км, смт. Рожнятів, Чеченське вдсх.	Дністер	р. Чечва	ЧЕР ДНЕСТР 1120 0039 0000 0000 0000 0016 01 26	✗
<input type="checkbox"/>	р. Чага, 66 км, с. Петровка, кордон з Молдовою	Річки Причорномор'я	р. Чага	ЧЕР КОГИЛЬ 0051 0000 0000 0000 0000 0066 01 51	✗
<input type="checkbox"/>	р. Циганська, 22 км, с. Мушкатівка, Мушкатівське вдсх,	Дністер	р. Циганська	ЧЕР ДНЕСТР 0872 0011 0000 0000 0000 0022 01 61	✗
<input type="checkbox"/>	р. Цата, 7 км, с. Ключи, Щорського р-ну (кордон з РФ)	Дніпро	р. Цата	ЧЕР ДНЕПР 0892 0228 0121 0000 0000 0007 01 74	✗
<input type="checkbox"/>	р. Хорол, 280 км, с. Панасовка, вище техн в/з м.Липова Долина, а/д міст	Дніпро	р. Хорол	ЧЕР ДНЕПР 0554 0112 0000 0000 0000 0280 01 59	✗
<input type="checkbox"/>	р. Хорол, 248 км, с. Лучки, Липоводолинського р-ну, кордон Сумської і Полтавської обл.	Дніпро	р. Хорол	ЧЕР ДНЕПР 0554 0112 0000 0000 0000 0248 02 59	✗
<input type="checkbox"/>	р. Хорол, 102 км, с. Вишняки, техн в/з села	Дніпро	р. Хорол	ЧЕР ДНЕПР 0554 0112 0000 0000 0000 0102 03 53	✗
<input type="checkbox"/>	р. Хорина, 0 км, ділянка від с. Павлівка до с.Травневе	Дон	р. Хорина	UA_M6.5.1_0421_01	✗

Рисунок 5.19 – Перелік зареєстрованих постів спостереження, по яким вже ведеться моніторинг

Натискаємо на кнопку «Редагування» потрібного поста спостереження. Відкриється сторінка з короткою інформацією про пост спостереження та таблицею «Таблиця спостережень», що зображена на рисунку 5.20. Додаємо новий рядок до таблиці, обираємо дату проведення спостережень і заповнюємо поля вимірними значеннями показників.



Таблиця спостережень

	Дата відбору проби	Гідробіологічні показники			Мікробіологічні показники			Цезій-137, пКі/дм3	Стронцій-90, пКі/дм3
		Фітопланктон, тис.клітин/дм3	Біомаса, мг/дм3	Хлорофіл, мг/дм3	Загальне мікробне число	Число лактозопозитивних кишкових паличок	Загальні коліформи (лактозопозитивні кишкові бактерії), індекс БГКП		
✓	02.07.2018		1						
✗	10.06.2009								
✗	12.02.2015								
✗	09.06.2015								
✗	22.09.2015								
✗	01.03.2016								
✗	03.11.2015								
✗	06.06.2016								
✗	21.09.2016								
✗	09.11.2016								
✗	21.03.2017								
✗	16.05.2017								

Рисунок 5.20 – Внесення вимірних значень показників

Для заповнення нормативів водогосподарського стану необхідно обрати пункт меню «Водогосподарська обстановка», підменю «Нормативи» і обрати потрібну категорію нормативів, наприклад, «Координати кривих об'ємів водосховищ». Це зображено на рисунку 5.21.



Рисунок 5.21 – Вибір категорії нормативів

Після цих дій відкриється сторінка із переліком водосховищ, для яких вже вказано нормативні значення. Для того, щоб додати нормативні значення



для нового водосховища, необхідно натиснути кнопку додавання нового запису, як зображено на рисунку 5.22.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Адмін Адмін Адмін

### Координати кривих об'ємів водосховищ

Фільтр по полям таблиці

Додати картку

Стор. 1 з 1

Перегляд 1 - 32 з 32

	Основний басейн	Водосховище	Проектний рівень, м	Проектний обсяг, млн.м3	
<input type="checkbox"/>	Р. РОСЬ	Білоцерківське нижнє	142,75	1,56	✖
<input type="checkbox"/>	Р. РОСЬ	Верхнє Білоцерківське	157,5	16,96	✖
<input type="checkbox"/>	Р. РОСЬ	Корсунь-Шевченківське	99,81	3,75	✖
<input type="checkbox"/>	Р. РОСЬ	Стеблівське	113,9	15,7	✖
<input type="checkbox"/>	Р. РОСЬ	Щербаківське	164	1,5	✖
<input type="checkbox"/>	Р. ДУНАЙ	Кагул	3,5	250,7	✖
<input type="checkbox"/>	Р. ДУНАЙ	Ялпуг	2,8	595	✖
<input type="checkbox"/>	Р. ДУНАЙ	Кугурлуй	2,8	293	✖
<input type="checkbox"/>	Р. ДУНАЙ	Катлабух	1,7	131	✖
<input type="checkbox"/>	Р. ДУНАЙ	Китай	1,5	111,9	✖
<input type="checkbox"/>	Р. ДУНАЙ	Сасик	0,2	500	✖
<input type="checkbox"/>	ВОДОСХОВИЩА КРИМУ	Чорноріченське	261	64,2	✖
<input type="checkbox"/>		Сімферопольське	292	36	✖

watermon.iisd.com.ua/CommonDocs/Docs/Add/29006

Рисунок 5.22 – Перелік доданих координат кривих об'ємів водосховищ

Заповнюємо всі необхідні значення і натискаємо кнопку «Зберегти», що зображено на рисунку 5.23.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Адмін Адмін Адмін

### Координати кривих об'ємів водосховищ

Зберегти Закрити

Основний басейн: \*  
Р. РОСЬ

Водосховище: \*  
Білоцерківське нижнє

Проектний рівень, м: 142,75  
Проектний обсяг, млн.м3: 1,56

Координати кривих об'ємів водосховища

Відмітка, м	Об'єм, млн.м3	Вільний об'єм, млн.м3	
143	1,73	-0,17	✖
142	1,0795	0,4805	✖
141	0,5756	0,9844	✖
140	0,1945	1,3655	✖
139	0	1,56	✖

Зберегти Закрити

Рисунок 5.23 – Додавання координат кривих об'ємів водосховища.

Сторінка додавання нормативу забору води на водогосподарських системах зображена на рисунку 5.24.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Админ Админ Админ

Норми заборів води на водогосподарських системах

Зберегти Закрити

Найменування каналу, зрошувальної системи: \*  
Ігулецька

Декадний план, тис.м3:  
9509

Сезони роботи

Сезон	Дата початку сезону	Дата кінця сезону
2018	13.04.2018	06.08.2018

Зберегти Закрити

Рисунок 5.24 – Додавання норм заборів води на водогосподарських системах

Сторінка додавання нормативу рівнів та витрат води на контрольних пунктах зображена на рисунках 5.25 – 5.26.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Админ Админ Админ

Нормативи рівнів та витрат води на контрольних пунктах

Зберегти Закрити

Річка-Пункт: \*  
Дунай - м. Кілія

«0» поста:  
-0,33

Норми витрат

Місяць	Норма витрат, м3/сек
червень	5040
травень	5470
січень	3770
серпень	3400
лютий	3990
листопад	2910
липень	4330
квітень	5390
жовтень	2790
грудень	3580

Перегляд 1 - 10 з 12

Рисунок 5.25 – Додавання нормативу рівнів та витрат на контрольних пунктах

Норми рівнів

	Число початку декади	Число кінця декади	Місяць	Максимальний рівень, м	Мінімальний рівень, м	
✎	21	31	січень	2,2	-0,4	✖
✎	21	31	лютий	2,3	-0,43	✖
✎	21	31	березень	2,4	-0,29	✖
✎	21	31	квітень	2,49	-0,11	✖
✎	21	31	травень	2,37	0,86	✖
✎	21	31	червень	2,22	-0,13	✖
✎	21	31	липень	2,17	-0,23	✖
✎	21	31	серпень	1,72	-0,31	✖
✎	21	31	вересень	1,62	-0,34	✖
✎	21	31	жовтень	1,5	-0,56	✖

Зберегти Закрити

Рисунок 5.26 – Продовження сторінки додавання нормативу рівнів та витрат на контрольних пунктах

Сторінка переліку нормативів гідрохімічних показників зображена на рисунках 5.27.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Админ Админ Админ

Нормативи гідрохімічних показників

Фільтр по полям таблиці

	Показник	Норма	
<input type="checkbox"/>	Жорсткість мг.екв./л	7	✖
<input type="checkbox"/>	Хлориди, мг/л	350	✖
<input type="checkbox"/>	Загальна мінералізація, мг/л	1000	✖

Рисунок 5.27 – Нормативи гідрохімічних показників

Для внесення даних по водогосподарському стану необхідно обрати пункт меню «Водогосподарська обстановка». Після цих дій відкриється сторінка із переліком дат досліджень водогосподарських станів, що зображена на рисунку 5.28.



МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Админ Админ Админ

Гідрологічна ситуація на водосховищах

Водогосподарські системи

Гідрологічна ситуація на контрольних пунктах

Гідрохімічна обстановка

Водогосподарська обстановка

Зберегти Закрити

Найменування каналу, зрошувальної системи	Забір води з початку сезону, тис.м3	Декадний план, тис.м3	Забір води з початку декади, тис.м3	Середньодобовий забір води, м3/с
Верхньо-Тарасівська				
Вольнянська				
Дніпро-Донбас				
Дніпро-Інгuleць				
Дніпро-Кривий Ріг	3870,72	6600	1693.44	2,8
Запорізька				
Інгuleцька				
Каховський	0	34200	0	0
Нікопольська				
Північно-Кримський				
Північно-Рогачицька				
Сіверський Донець-Донбас	19906,56	15400	8709.12	14,4
у т.ч. в Сіверський Донець				
у т.ч. до Харкова	2211,84	1000	967.68	1,6
у т.ч. подача до Криму				
Фрунзенська				

Рисунок 5.30 – Внесення вимірних показників по водогосподарських системам

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти  
Админ Админ Админ

Гідрологічна ситуація на водосховищах

Водогосподарські системи

Гідрологічна ситуація на контрольних пунктах

Гідрохімічна обстановка

Водогосподарська обстановка

Зберегти Закрити

Гідрологічна обстановка

Річка-Пункт

Внесення запису

Річка-Пункт: \*

Дніпро - м. Київ

Фактичний рівень (за даними Укргідрометцентру): \*

6

Фактична витрата, м3/сек: \*

3

Зберегти Закрити

Фактична витрата, м3/сек

Закрити

Рисунок 5.31 - Внесення вимірних показників для гідрологічної ситуації на контрольних пунктах.

Рисунок 5.32 - Внесення вимірних показників для гідрохімічної обстановки на контрольних пунктах.

Ввійшовши в систему з правами адміністратора, можна переглянути всі доступні звіти, що наведені на рисунках 5.33 – 5.34. Для цього треба перейти в пункт меню «Звітність».

Рисунок 5.33 – Перелік доступних звітів

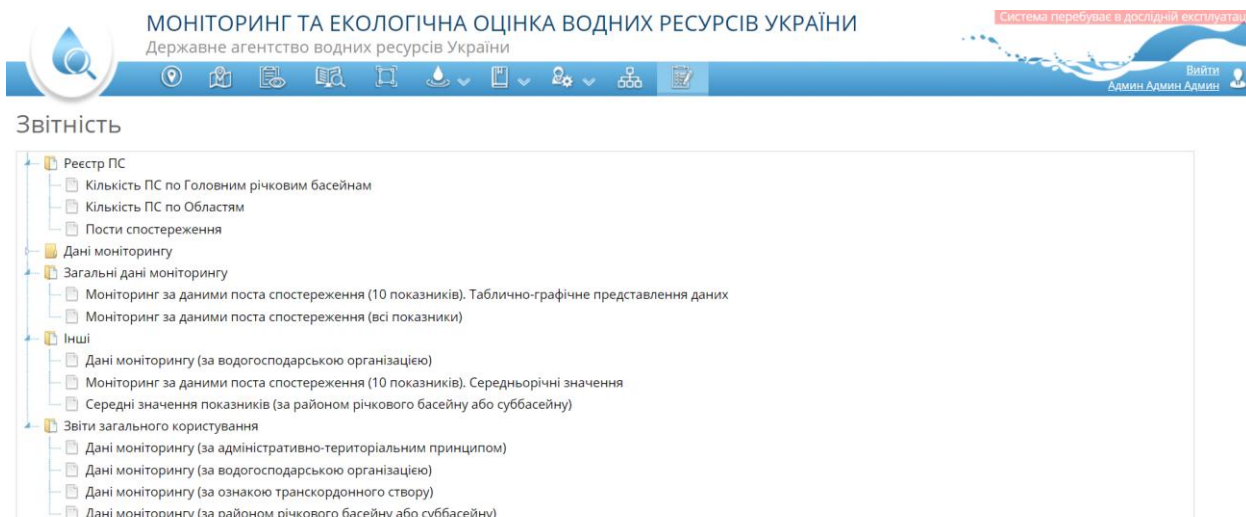


Рисунок 5.34 – Продовження переліку доступних звітів

Для того, щоб зареєструвати користувача, необхідно вибрати пункт меню «Адміністрування» та із випадаючого списку обрати «Керування користувачами», як показано на рисунку 5.35.

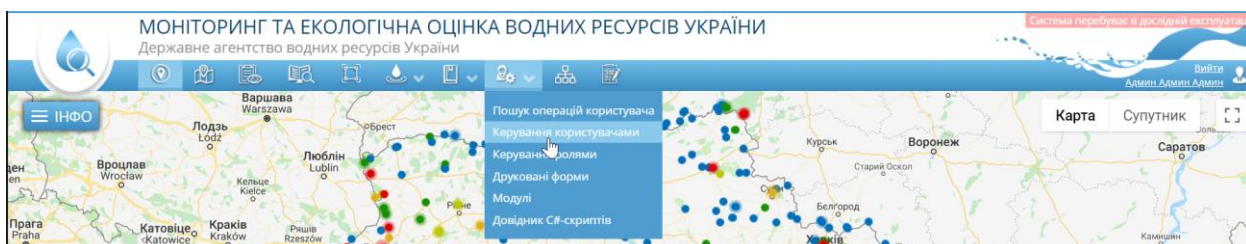


Рисунок 5.35 – Перехід на сторінку «Керування користувачами»

На сторінці, що відкрилася, натискаємо кнопку «Додати нового користувача», як показано на рисунку 5.36.

Додати нового користувача

			Ролі користувача	Прізвище	Посада	Установа	Зареєстрований	Останнє підключення	
			admin	- Адміністратор системи моніторингу	Адміністратор		Державне агентство водних ресурсів України	16.12.2015 13:05:57	01.11.2018
			admin1	- Системний адміністратор	Адмін	Адміністратор		06.10.2014 9:09:00	11.05.2019
			BUVR_DESNA	- Користувач підрозділу Водагентства	Десянянське		Десянянське БУВР	13.12.2017 12:40:41	02.10.2018
			BUVR_DNIPRO	- Користувач підрозділу Водагентства	Дніпровське		Дніпровське БУВР	13.12.2017 12:37:48	22.10.2018
			BUVR_DNISTER	- Користувач підрозділу Водагентства	БУВР Пруту		БУВР Пруту та Сірету	13.12.2017 13:16:01	01.11.2018
			BUVR_DUNAJ	- Користувач підрозділу Водагентства	Дунайське		Дунайське БУВР	13.12.2017 13:24:28	01.11.2018
			BUVR_PBUG	- Користувач підрозділу Водагентства	Південно-Бузьке		Південно-Бузьке БУВР	13.12.2017 13:10:40	22.10.2018
			BUVR_ROSI	- Користувач підрозділу Водагентства	БУВР		БУВР р. Рось	13.12.2017 12:13:59	30.10.2018
			BUVR_SD	- Користувач підрозділу Водагентства	Сіверсько-Донецьке		Сіверсько-Донецьке БУВР	13.12.2017 12:42:28	26.10.2018
			BUVR_TYSY	- Користувач підрозділу Водагентства	БУВР		БУВР Тиси	13.12.2017 13:25:39	08.10.2018

<

Рисунок 5.36 – Реєстрація нового користувача



На сторінці, що відкрилася, заповнюємо всі необхідні поля і натискаємо кнопку «Зберегти», як показано на рисунку 5.37.

### Додання користувача

Системні дані

Логін *	Активний
Test	Так ▾
Пароль: *	Повторіть пароль: *
.....	.....

Особисті дані

Прізвище *	
Іванов	
Ім'я *	По батькові
Іван	Іванович
Посада	
Директор департаменту	
Email *	Телефон
ivan@gmail.com	111111
Установа	
Державне агентство водних ресурсів України ▾	

Ролі користувача

<input type="checkbox"/> Користувач модуля водогосподарської обстановки <input type="checkbox"/> Системний адміністратор <input checked="" type="checkbox"/> Публічний користувач <input type="checkbox"/> Користувач підрозділу Водагентства <input type="checkbox"/> Адміністратор системи моніторингу
--

Параметри плагіну 'Документи'

Мова інтерфейсу
Українська ▾

Зберегти Скасувати

Рисунок 5.37 – Сторінка реєстрації користувача

Для заповнення довідників необхідно обрати пункт меню «Загальні довідники», як показано на рисунку 5.38.

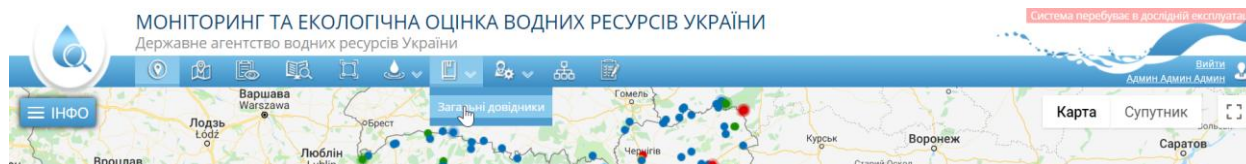


Рисунок 5.38 – Перехід на сторінку керування довідниками

На сторінці в поле пошуку вводимо назву потрібного словника. Натискаємо кнопку «Пошук». Обираємо знайдений словник із таблиці

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		99



«Довідники». В таблицю «Значення довідника» завантажаться відповідні значення словника. Натиснувши на кнопку «Додати значення довідника», що розташована над таблицею, можна, відповідно, додати нове значення довідника. Також можна відредагувати вже існуючі значення. Це показано на рисунку 5.39.

МОНІТОРИНГ ТА ЕКОЛОГІЧНА ОЦІНКА ВОДНИХ РЕСУРСІВ УКРАЇНИ  
Державне агентство водних ресурсів України

Система перебуває в дослідній експлуатації

Вийти

Адмін Адмін Адмін

НДІ

Назва довідника  
Статус поста спостереження

Назва атрибута довідника

Пошук Очистити

Довідники

Назва довідника	Назва атрибута довідника
Статус поста спостереження	Назва статусу ПС

Значення довідника

Повне значення (Українська)	Коротке значення (Українська)	Батьківський запис	Початок актуальності	Кінець актуальності	Локальний код	Порядок сортування
Діючий	Діючий		10.11.2017			
Не діючий	Не діючий		10.11.2017			
Законсервованій	Законсервованій		10.11.2017			

Рисунок 5.39 – Заповнення довідника

## 5.2 Випробування програмного продукту

### 5.2.1 Мета випробувань

Метою випробувань являється перевірка відповідності функцій комплексу задач підтримки процесу проведення моніторингу та екологічної оцінки водних ресурсів України вимогам технічного завдання.

### 5.2.2 Загальні положення

Випробування проводяться на основі наступних документів:

- ГОСТ 34.603–92. Інформаційна технологія. Види випробувань автоматизованих систем;
- ГОСТ РД 50-34.698-90. Автоматизовані системи вимог до змісту документів.

### 5.2.3 Результати випробувань

Під час тестування була перевірена основна функціональність комплексу задач (КЗ). В таблицях 5.1-5.2 наведено декілька випробувань для перевірки функції «Вхід в систему». Перелік випробувань інших основних функціональних можливостей наведено в документі Програма та методика випробувань.

Таблиця 5.1 – Вхід в систему

Мета тесту	Перевірка функції «Вхід в систему»
Початковий стан КЗ	Відкрита сторінка для входу в систему
Вхідні данні	Логін та пароль користувача
Схема проведення тесту	Ввести у поле «Логін» логін користувача, а у поле «Пароль» - пароль користувача. Натиснути кнопку «Вхід»
Очікуваний результат	Відкрита головна сторінка із пунктами меню із врахуванням прав доступу поточного користувача
Стан КЗ після проведення випробувань:	Відкрита головна сторінка із пунктами меню із врахуванням прав доступу поточного користувача

Таблиця 5.2 – Перевірка заповнення логіна і пароля

Мета тесту	Перевірка функції «Перевірка заповнення логіна і пароля»
Початковий стан КЗ	Відкрита сторінка для входу в систему
Вхідні данні	Логін або пароль
Схема проведення тесту	Ввести у поле «Логін» логін користувача, а поле «Пароль» - залишити пустим. Або навпаки. Натиснути кнопку «Вхід»

## Продовження таблиці 5.2

Мета тесту	Перевірка функції «Перевірка заповнення усіх атрибутів»
Очікуваний результат	Кнопка «Вхід» недоступна для натискання. Біля пустих полів повідомлення валідації «Поле обов'язкове для заповнення». Вхід в систему не виконано
Стан КЗ після проведення випробувань	Відкрита сторінка для входу в систему

**Висновок до розділу**

В даному розділі представлено керівництво користувача. В ньому наведена детальна інструкція щодо користування системою. Всі кроки проілюстровано екранними формами веб-системи. Описано всі послідовності дій відповідно до функціональних вимог.

Наведено опис тестів та порядок їх виконання для перевірки відповідності функцій комплексу задач підтримки процесу проведення моніторингу та екологічної оцінки водних ресурсів України функціональним вимогам, що представлені у технічному завданні.

## ЗАГАЛЬНІ ВИСНОВКИ

У даному дипломному проекті розглядається проблема забруднення водних ресурсів України та контролю стану водогосподарської обстановки. В Україні головним державним виконавчим органом, що здійснює державну політику у сфері розвитку водного господарства, контролю, управління та відновлення водних ресурсів виступає Держводагенство.

Було досліджено процеси діяльності та взаємодії Держводагенства та підлеглих водогосподарських організацій і виявлено необхідність створення системи для підтримки процесів проведення моніторингу та екологічної оцінки водних ресурсів України. Дана розробка необхідна для покращення інформаційного супроводу робіт із здійснення моніторингу, підвищення оперативності та ефективності аналізу наявої інформації про стан водних ресурсів, спрощення регулярної актуалізації та багатоцільового використання інформації про стан водних ресурсів.

Описано функціональну модель системи, що охоплює всі необхідні процеси та дії вищезгаданих організацій. Розроблено інформаційне забезпечення системи. Система має великий перелік звітів в якості вихідних даних, що значно полегшує роботу працівників при оформленні звітності і отриманні підсумкових даних для подальшого аналізу та прийняття рішень.

Використання геоінформаційних технологій в даній розробці дає можливість якісної візуалізації інформації про розміщення постів спостереження та про перевищення гранично допустимої концентрації показників на постах спостереження. Завдяки використанню маркерів різних кольорів та розмірів на мапі можна швидко оцінити загальний стан водних ресурсів як всієї України, так і окремих її регіонів та дати відповідну екологічну оцінку.

У математичному розділі було досліджено методи проведення ефективної екологічної оцінки водних ресурсів України, зокрема, методи прогнозування. У результаті проведеної роботи було реалізовано 4

					ДП ІС-5109.1181-с.ПЗ	Арк.
						103
Змн.	Арк.	№ докум.	Підпис	Дата		

найпоширеніші методи прогнозування – метод ковзного середнього, метод експоненціального згладжування, метод найменших квадратів та метод прогнозування з використанням нейронної мережі. Для прогнозування використовувалися дані моніторингу показника забрудненості амоній-іони за 25 років. Після проведення експериментальних досліджень і обчислення критерію якості прогнозування – середньоквадратичного відхилення, було отримано результати, що свідчать про те, що нейронна мережа дає прогноз з найвищою точністю серед усіх використаних методів.

Інформаційна система моніторингу та екологічної оцінки водних ресурсів України успішно пройшла всі випробування і є впровадженою для використання у Держводагенстві та інших водогосподарських організаціях.

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		104

## ПЕРЕЛІК ПОСИЛАНЬ

1. Казмірчук А. В. Дослідження методів проведення ефективної екологічної оцінки водних ресурсів України / Аліна Василівна Казмірчук. // INNOVATIVE SOLUTIONS IN MODERN SCIENCE. – 2019. – №30. – С. 5–17.
2. AQUARIUS WebPortal [Електронний ресурс] – Режим доступу до ресурсу: <https://aquaticinformatics.com/products/aquarius-webportal/>.
3. Version LEADS - Environmental Monitoring System Software (EMS) [Електронний ресурс] – Режим доступу до ресурсу: <https://www.environmental-expert.com/software/version-leads-environmental-monitoring-system-software-ems-514983>.
4. Відкриті дані про якість води в Україні [Електронний ресурс] – Режим доступу до ресурсу: [http://texty.org.ua/pg/news/hohobi/read/85483/Vidkryti\\_dani\\_pro\\_jakist\\_vody\\_v\\_Ukrajini](http://texty.org.ua/pg/news/hohobi/read/85483/Vidkryti_dani_pro_jakist_vody_v_Ukrajini).
5. Держводагенство офіційний сайт [Електронний ресурс] – Режим доступу до ресурсу: <https://www.davr.gov.ua>.
6. Екологічний моніторинг довкілля [Електронний ресурс] – Режим доступу до ресурсу: <https://menr.gov.ua/content/ekologichniy-monitoring-dovkillya.html>.
7. Экологический мониторинг Калужской области [Електронний ресурс] – Режим доступу до ресурсу: <http://www.ecoanalyt.ru/monitoring/index.html>.
8. Водне господарство [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/Водне\\_господарство](https://uk.wikipedia.org/wiki/Водне_господарство).
9. Часові ряді, методи аналізу часових рядів [Електронний ресурс] – Режим доступу до ресурсу: <https://posibnyky.vntu.edu.ua/kocuba/p4.html>.

- 10.Новиков В.А. Организация и обучение искусственных нейронных сетей: Экспериментальное учеб. пособие. / В.А.Новиков, Л.В.Калацкая, В.С.Садов – Минск: БГУ, 2003. – 72 с.
- 11.Метод ковзного середнього [Електронний ресурс] – Режим доступу до ресурсу: [https://wiki.tntu.edu.ua/Метод\\_ковзного\\_середнього](https://wiki.tntu.edu.ua/Метод_ковзного_середнього).
- 12.Розробка прогнозу за допомогою методу найменших квадратів [Електронний ресурс] – Режим доступу до ресурсу: <http://www.ekonomika-st.ru/drugie/metodi/metodi-prognoz-1-5.html>.
- 13.Вибір параметрів багатосарових нейронних мереж прямого розповсюдження [Електронний ресурс] – Режим доступу до ресурсу: [http://mei06.narod.ru/sem7/iis/shpora/page2\\_9.htm](http://mei06.narod.ru/sem7/iis/shpora/page2_9.htm).
- 14.Хайкин С. Нейронные сети / С. Хайкин. – М. : Вильямс, 2006. – 1103 с
- 15.Ставицький О. В. Використання нейронних мереж для прогнозування у фінансовій сфері [Електронний ресурс] / О. В. Ставицький, М. О. Мозолевська – Режим доступу до ресурсу: <http://ape.fmm.kpi.ua/article/view/102584>.
- 16.Прогнозування за допомогою нейронних мереж [Електронний ресурс] – Режим доступу до ресурсу: [https://wiki.tntu.edu.ua/Прогнозування\\_за\\_допомогою\\_нейронних\\_мереж](https://wiki.tntu.edu.ua/Прогнозування_за_допомогою_нейронних_мереж).
- 17.Грофф Д. Р. SQL: полное руководство / Д. Р. Грофф, П. Н. Вайнберг., 2018. – 960 с.
- 18.Google Maps Platform Documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://developers.google.com/maps/documentation/>.
- 19.Преваги платформи .NET [Електронний ресурс] – Режим доступу до ресурсу: [https://studopedia.ru/1\\_84531\\_preimushchestva-platformi-NET.html](https://studopedia.ru/1_84531_preimushchestva-platformi-NET.html).
- 20.Фриман А. ASP.NET MVC 4 / Адам Фриман., 2014. – 688 с.

21. Microsoft SQL Server [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](https://uk.wikipedia.org/wiki/Microsoft_SQL_Server).

					ДП ІС-5109.1181-с.ПЗ	Арк.
						107
Змн.	Арк.	№ докум.	Підпис	Дата		



## Додаток А

**Тексти програмного коду****Інформаційна система моніторингу та екологічної оцінки водних  
ресурсів України**

---

(Найменування програми (документа))

---

*DVD-R*

(Вид носія даних)

---

*25 арк, 4 Гб*

(Обсяг програми (документа) , арк.,) Кб)

Київ – 2019 року

					ДП ІС-5109.1181-с.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		108

**GDKMapController.cs**

```

public class GDKMapController : Controller
{
    public UserInfo userInfo;
    private IGDKMapService service;
    public GDKMapController(IGDKMapService service, IUserDataProvider  userDataProvider)
    {
        this.userInfo = userDataProvider.GetCurrentUserInfo();
        this.service = service;
    }

    public ActionResult Index()
    {
        if (userInfo != null)
        {
            ViewBag.UserNick = userInfo.FullUserName;
            ViewBag.IsAuthenticated = userInfo.IsAuthenticated;
        }
        return View();
    }

    public JsonResult GetResult()
    {
        var result = service.GetLastData();
        return Json(result, JsonRequestBehavior.AllowGet);
    }
}

```

**MapEcoWaterMonController.cs**

```

public class MapEcoWaterMonController : Controller
{
    private readonly IMapEcoWaterMonService service;
    private IMembershipService membershipService;
    public UserInfo userInfo;
    private IFormsAuthenticationService formsService;
    public MapEcoWaterMonController(IMapEcoWaterMonService service, IMembershipService
membershipService, IUserDataProvider userDataProvider, IFormsAuthenticationService formsService)
    {
        this.service = service;
        this.userInfo = userDataProvider.GetCurrentUserInfo();
        this.membershipService = membershipService;
        this.formsService = formsService;
    }

    public ActionResult Index()
    {
        IEnumerable<ListModel> dicList = service.GetDicList();

        ViewData["Laboratory"] = dicList.Where(x => x.id == 29003).Select(x =>
            new SelectListItem()
            {
                Text = x.Text.ToString(),
                Value = x.Value.ToString()
            }).ToList();

        ViewData["RiverBas"] = dicList.Where(x => x.id == 29001).Select(x =>
            new SelectListItem()

```

```

        {
            Text = x.Text.ToString(),
            Value = x.Value.ToString()
        }).ToList();

ViewData["WaterCompName"] = dicList.Where(x => x.id == 29005).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();
ViewData["WaterDepName"] = dicList.Where(x => x.id == 29007).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();
ViewData["OPType"] = dicList.Where(x => x.id == 29023).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();

ViewData["PeriodOfCont"] = dicList.Where(x => x.id == 29025).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();
ViewData["MonType"] = dicList.Where(x => x.id == 29021).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();
ViewData["NormType"] = dicList.Where(x => x.id == 29019).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();
ViewData["TransOP"] = dicList.Where(x => x.id == 29027).Select(x =>
    new SelectListItem()
    {
        Text = x.Text.ToString(),
        Value = x.Value.ToString()
    }).ToList();

if (userInfo != null)
{
    ViewBag.UserNick = userInfo.FullUserName;
    ViewBag.IsAuthenticated = userInfo.IsAuthenticated;
}

return View();
}

```

```

public MvcHtmlString FilterMap(int?[] RiverBas, int?[] Laboratory, int?[] WaterCompName, int?[]
WaterDepName, int?[] OPType, int?[] PeriodOfCont, int?[] MonType, int?[] NormType, int?[] TransOP)

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						110
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    decimal?[] RiverBasDec = Array.ConvertAll(RiverBas, x => (decimal?)x);
    decimal?[] LaboratoryDec = Array.ConvertAll(Laboratory, x => (decimal?)x);
    decimal?[] WaterCompNameDec = Array.ConvertAll(WaterCompName, x => (decimal?)x);
    decimal?[] WaterDepNameDec = Array.ConvertAll(WaterDepName, x => (decimal?)x);
    decimal?[] OPTTypeDec = Array.ConvertAll(OPTType, x => (decimal?)x);
    decimal?[] PeriodOfContDec = Array.ConvertAll(PeriodOfCont, x => (decimal?)x);
    decimal?[] MonTypeDec = Array.ConvertAll(MonType, x => (decimal?)x);
    decimal?[] NormTypeDec = Array.ConvertAll(NormType, x => (decimal?)x);
    decimal?[] TransOPDec = Array.ConvertAll(TransOP, x => (decimal?)x);

    var list = service.FilterMap(RiverBasDec, LaboratoryDec, WaterCompNameDec, WaterDepNameDec,
    OPTTypeDec, PeriodOfContDec, MonTypeDec, NormTypeDec, TransOPDec);
    var serializer = new JavaScriptSerializer();
    var result = new MvcHtmlString(serializer.Serialize(list));
    return result;
}

public JsonResult GetChilds(int?[] WaterCompName)
{
    decimal?[] WaterCompNameDec = Array.ConvertAll(WaterCompName, x => (decimal?)x);
    Dictionary<string, string> d = service.GetChilds(WaterCompNameDec);
    return Json(d, JsonRequestBehavior.AllowGet);
}

[AllowAnonymous]
public ActionResult RedirectToReport(ReportParams reportParams)
{
    var value = Configurator.NotAuthorizeUser;
    var user = membershipService.GetUser(value);
    if (User.Identity.Name == "" && user != null && user.IsApproved != false)
    {
        formsService.SignIn(value, true);
    }

    return RedirectToAction("ViewReportParams", "ViewReports", reportParams);
}
}

```

### ParsingDocFileController.cs

```

public class ParsingDocFileController : Controller
{
    private readonly IParsingDocFileService _service;
    public ParsingDocFileController(IParsingDocFileService service)
    {
        _service = service;
    }
    public PartialViewResult GetForm(decimal dossierId)
    {
        return PartialView("PartialFileLoad", dossierId);
    }

    [HttpPost]
    public JsonResult OpenFile(decimal dossierId)
    {
        System.Web.HttpPostedFileBase file = Request.Files["fileSource"];

        if (file != null && file.ContentLength > 0 && file.InputStream != null)
        {

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						111
Змн.	Арк.	№ докум.	Підпис	Дата		

```

return Json(_service.ParseDoc(file.InputStream, dossierId), JsonRequestBehavior.AllowGet);

    }
    else
    {
        return Json("", JsonRequestBehavior.AllowGet);
    }

    public JsonResult DeleteRecords(decimal dossierId, decimal sectionId, decimal docCategory, bool
saveRowDB)
    {
        return Json(_service.DeleteRecords(dossierId, sectionId, docCategory, saveRowDB),
JsonRequestBehavior.AllowGet);
    }
}

```

### PostsController.cs

```

public class PostsController : Controller
{
    private IDossierRepository repository;
    private IPostsService postsService;
    public PostsController(IDossierRepository repository, IPostsService postsService)
    {
        this.repository = repository;
        this.postsService = postsService;
    }
    public JsonResult GetIndexes(string sidx, string sord, int page, int rows, string nodeid, string parentid, int?
n_level)
    {
        var parentList = (from dv in repository.Table<Dictionary_Value>()
join dv2 in repository.Table<Dictionary_Value>() on 1 equals 1
where dv.DIC_IDP == 29029
&& (dv2.DICV_LID == dv.DICV_IDP || dv2.DICV_IDP == dv.DICV_IDP)
&& dv.DICV_EDAT >= DateTime.Now
select new
{
    //id = dv.DICV_LID == null ? dv.DICV_UID.ToString() : dv.DICV_IDP.ToString() + "_" +
dv.DICV_LID.ToString(), //доделать утром
    name = dv2.DICV_LNAM,
    dicv_idp = dv2.DICV_LID == null ? dv2.DICV_IDP.ToString() : dv2.DICV_LID.ToString() +
"_" + dv2.DICV_IDP.ToString(), //доделать утром
    dicv_lid = dv2.DICV_LID != null ? dv2.DICV_LID : 0,
    level = dv2.DICV_LID != null ? 1 : 0
}).ToList().OrderBy(x => x.dicv_idp).ThenBy(x => x.name);

        var jsonData = new
        {
            total = 1,
            page = 1,
            records = parentList.Count(),
            rows = (
                from pl in parentList
                select new
                {
                    i = pl.dicv_idp,
                    cell = new object[]
                    {
                        pl.name,
                        "",

```

```

        false,
        pl.dicv_idp,
        pl.name,
        pl.level,
        pl.dicv_lid,
        pl.dicv_lid != 0 ? "true" : "false",
        (pl.level == 0)?"true":"false"
    }
    }).ToArray()
};
return Json(jsonData, JsonRequestBehavior.AllowGet);
// return null; /*Json(jsonData, JsonRequestBehavior.AllowGet);*/
}

public JsonResult DisableFields(decimal dossierId)
{
    if (dossierId != 0)
    {
        var disabledFields = (from dl in repository.Table<Dossier_List>()
                               join dl2 in repository.Table<Dossier_List>() on dl.DLS_IDP equals dl2.DLS_LID
                               join dv in repository.Table<Dossier_Value>() on dl2.DLS_IDP equals dv.DLS_IDP
                               join gv in repository.Table<Grid_Value>() on dv.DVL_IDP equals gv.DVL_IDP
                               where dl.DLS_DID == 29002 && dl.DLS_DDAT == null &&
                                     dv.DVL_NUM1 == dossierId && gv.GRD_LID == 29048 &&
                                     gv.GRD_NUM3 != null
                               select new
                               {
                                   index = Convert.ToInt32(gv.GRD_NUM1).ToString() + "_" +
                                   Convert.ToInt32(gv.GRD_NUM2).ToString()
                               }).GroupBy(x => x.index).ToList();

        List<string> result = new List<string>();
        foreach (var row in disabledFields)
        {
            result.Add(row.Key);
        };

        return Json(result, JsonRequestBehavior.AllowGet);
    }
    return null;
}

public void InsertGridValue(List<TreeModel> arr, decimal dossierId, decimal sectionId)
{
    postsService.InsertGridValue(arr, dossierId, sectionId);
}

public JsonResult GetGridData(decimal dossierId, decimal sectionId)
{
    var result = postsService.GetGridData(dossierId, sectionId);
    return Json(result, JsonRequestBehavior.AllowGet);
}
}

```

### UnauthorizedController.cs

```

public class UnauthorizedController : Controller
{
    public IFormsAuthenticationService FormsService { get; set; }
}

```

					ДП IC-5109.1181-с.ПЗ	Арк.
						113
Змн.	Арк.	№ докум.	Підпис	Дата		

```

private IMembershipService _MembershipService;
private IUserDataProvider _userDataProvider { get; set; }
public UserInfo userInfo;

public UnauthorizedController(IFormsAuthenticationService formsService,
    IUserDataProvider userDataProvider, IMembershipService MembershipService)
{
    FormsService = formsService;
    _userDataProvider = userDataProvider;
    this.userInfo = userDataProvider.GetCurrentUserInfo();
    _MembershipService = MembershipService;
}

[HttpGet]
public ActionResult LogOnWithSystemLogin(string Login)
{
    if (userInfo.FullUserName == string.Empty)
    {
        string value = null;
        if (Login == null)
            value = Configurator.NotAuthorizeUser;
        else
            value = Login;

        var user = _MembershipService.GetUser(value);
        if (user != null && user.IsApproved != false)
        {
            if (userInfo != null)
            {
                FormsService.SignOut();
                Session.Clear();
            }
            FormsService.SignIn(value, false);
            _userDataProvider.GetCurrentUserInfo();
        }
        else
            throw new Exception("Не правильные набор параметров.");
    }
    return RedirectToAction("List", "ViewReports");
}

[HttpGet]
public ActionResult SignOut()
{
    if (userInfo.UserName == "SystemUserRoleCitizen")
    {
        FormsService.SignOut();
    }
    return RedirectToAction("Account", "LogOn");
}
}

```

**WaterResourcesController.cs**

```

public class WaterResourcesController : Controller
{
    private IDossierRepository repository;
    private IWaterResourcesService waterService;
    private ITableService _service;

```

					ДП IC-5109.1181-с.ПЗ	Арк.
						114
Змн.	Арк.	№ докум.	Підпис	Дата		

```

private IDossierAttributeManager _dosAttrManager;
public WaterResourcesController(IDossierRepository repository, IWaterResourcesService waterService,
ITableService _service, IDossierAttributeManager _dosAttrManager)
{
    this.repository = repository;
    this.waterService = waterService;
    this._service = _service;
    this._dosAttrManager = _dosAttrManager;
}
public JsonResult ChangeGridAndValues(decimal attr29034, bool editType)
{
    var result = waterService.ChangeGridAndValues(attr29034, editType);
    return Json(result, JsonRequestBehavior.AllowGet);
}

[ValidateInput(false)]
public JsonResult EditRow(decimal dossierId, decimal sectionId, string jsonObjectString)
{
    //формируем объект с данными который пришел из клиента
    Dictionary<string, string> values = new JavaScriptSerializer().Deserialize<Dictionary<string,
string>>(jsonObjectString);
    var grdAttr = (DataTable)_dosAttrManager.LoadAttribute(dossierId, sectionId, 29048, false, 29003).Value;
    List<DossierAttribute> list = new List<DossierAttribute>();
    DateTime date = Convert.ToDateTime(values["GRD_DATE1"]);

    values.Remove("Id");
    values.Remove("GRD_DATE1");
    values.Remove("edit");
    values.Remove("delete");

    //выбираем из списка записей только те которые относятся к нашей строке, по этому делаем поиск по
    date
    var isExistRow = grdAttr.AsEnumerable().Where(x => x.Field<DateTime>("29049") == date);
    List<DataRow> deleteList = new List<DataRow>();

    if (isExistRow.Count() != 0)
    {
        foreach (var item in isExistRow)
        {
            //формируем индекс по которому произведем поиск по массиву с данными
            string index = Convert.ToInt32(item["29050"]).ToString() + "_" +
            Convert.ToInt32(item["29051"]).ToString();
            //поиск по индексу
            var dicItem = values.Where(x => x.Key == index).First();
            if (dicItem.Value != string.Empty)
            {
                //если значение показателя найденной ячейки не пустое - записываем его
                item.SetField("29052", Convert.ToDouble(dicItem.Value.Replace(".", ",")));
                //выставляем флаг isChg чтобы при сохранении карточки, общий функционал подхватил наш
                блок
                item.SetField("isChg", 1);
                //удаляем из массива с данными только что добавленное значение
                values.Remove(dicItem.Key);
            }
            else
            {
                //если значение показателя пустое - значит его удалили, соответственно удаляем строку из
                Grid_Value
                deleteList.Add(item);
            }
        }
    }
}

```

					ДП IC-5109.1181-с.ПЗ	Арк.
						115
Змн.	Арк.	№ докум.	Підпис	Дата		



```

        foreach (var row in deleteList)
        {
            grdAttr.Rows.Remove(row);
        }
    }

    //выбираем оставшиеся не пустые значения которые к которым не нашли пары (значит их только что
добавили)
    values = values.Where(x => x.Value != string.Empty).ToDictionary(x => x.Key, x => x.Value);
    if (values.Count != 0)
    {
        //получаем минимальный идентификатор
        int rowId = 0;
        foreach (DataRow dr in grdAttr.Rows)
        {
            int accountLevel = Convert.ToInt32(dr["Id"]);
            rowId = Math.Min(rowId, accountLevel);
        }
        rowId = rowId > 0 ? -1 : --rowId;
        foreach (var item in values)
        {
            DataRow row = grdAttr.NewRow();
            row["Id"] = rowId--;
            row["29049"] = date;
            row["29050"] = item.Key.Substring(0, item.Key.IndexOf("_"));
            row["29051"] = item.Key.Substring(item.Key.IndexOf("_") + 1);
            if (item.Value != string.Empty)
                row["29052"] = Convert.ToDouble(item.Value.Replace(".", ","));
            else
                row["29052"] = DBNull.Value;

            grdAttr.Rows.Add(row);
        }
    }
    return null;
}

public JsonResult ListData(JqGridPostData gridHeader, string headerList, decimal dossierId, decimal
sectionId, bool firstLoad)
{
    //формируем хедер для того чтобы выставить в колонку соответствующее значение
    List<string> tempArray = new JavaScriptSerializer().Deserialize<List<string>>(headerList);
    int totalCount = 0;
    object res = null;

    //получаем список значений
    var todoListsResults = waterService.GetListData(dossierId, sectionId, firstLoad);

    //выполняем группировку по дате
    var groupingList = todoListsResults.GroupBy(x => x.GRD_DATE1);
    IEnumerable<ListDataModel> tempData;

    var result = new List<List<string>>>();
    int i = -1;
    foreach (var item in groupingList)
    {
        //для каждой даты формируем строку, длиной нашего хедера
        List<string> tempResult = Enumerable.Repeat(string.Empty, tempArray.Count).ToList();
        //выбираем из списка значений только которые относятся к текущей дате (item.key)
        tempData = todoListsResults.Where(x => x.GRD_DATE1 == item.Key);
    }

```

					ДП IC-5109.1181-с.ПЗ	Арк.
						116
Змн.	Арк.	№ докум.	Підпис	Дата		

```

//далее идет заполнение полей по индексам (чтобы в каждой колонке было соответствующее ей
значение)
tempResult[tempArray.IndexOf("Id")] = (i--).ToString();
tempResult[tempArray.IndexOf("GRD_DATE1")] = item.Key.Value.Date.ToString("dd.MM.yyyy");
foreach (var tempItem in tempData)
{
    //для значений показателя формируем идентификатор по которому определяем индекс
    string index = Convert.ToInt32(tempItem.GRD_NUM1).ToString() + "_" +
    Convert.ToInt32(tempItem.GRD_NUM2).ToString();
    //по индексу ищем ячейку и вставляем в неё значение
    tempResult[tempArray.IndexOf(index)] = tempItem.GRD_NUM3 != null ?
    Convert.ToDouble(tempItem.GRD_NUM3).ToString() : string.Empty;
}

result.Add(tempResult);
}

#region сортування
if (gridHeader.sidx == "Id")
    result = gridHeader.sord == "asc" ? result.OrderBy(x =>
    Convert.ToInt32(x[tempArray.IndexOf("Id")])).ToList() : result.OrderByDescending(x =>
    Convert.ToInt32(x[tempArray.IndexOf("Id")])).ToList();
else
    result = gridHeader.sord == "asc" ? result.OrderBy(x =>
    Convert.ToDateTime(x[tempArray.IndexOf("GRD_DATE1")])).ToList() : result.OrderByDescending(x =>
    Convert.ToDateTime(x[tempArray.IndexOf("GRD_DATE1")])).ToList();
#endregion

totalCount = result.Count;

if (gridHeader.rows != 999999 && totalCount > gridHeader.rows) //если выбрано не "Всі"
    result = result.Skip((gridHeader.page - 1) * gridHeader.rows).Take(gridHeader.rows).ToList();
else
    gridHeader.page = 1;

res = gridHeader.MakeJsonResult(totalCount, result);
return Json(res, JsonRequestBehavior.AllowGet);
}

public JsonResult DeleteRow(decimal dossierId, decimal sectionId, string date)
{
    waterService.DeleteRow(dossierId, sectionId, date);
    return null;
}

public JsonResult FillDateArray(decimal dossierId, decimal sectionId)
{
    return Json(waterService.FillDateArray(dossierId, sectionId), JsonRequestBehavior.AllowGet);
}

public static void AddProperty(ExpandoObject expando, string propertyName, object propertyValue)
{
    // ExpandoObject supports IDictionary so we can extend it like this
    var expandoDict = expando as IDictionary<string, object>;

    if (expandoDict.ContainsKey(propertyName))
        expandoDict[propertyName] = propertyValue;
    else
        expandoDict.Add(propertyName, propertyValue);
} }

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						117
Змн.	Арк.	№ докум.	Підпис	Дата		

**WaterSituationController.cs**

```

public class WaterSituationController : Controller
{
    private IWaterSituationService waterSituationService;
    private IDossierAttributeManager dosAttrManager;

    public WaterSituationController(IWaterSituationService waterSituationService, IDossierAttributeManager
dosAttrManager)
    {
        this.waterSituationService = waterSituationService;
        this.dosAttrManager = dosAttrManager;
    }

    public JsonResult GetMonth( decimal dictionaryId, string localCode)
    {
        var result = waterSituationService.GetMonth(dictionaryId, localCode);
        return Json(result, JsonRequestBehavior.AllowGet);
    }

    public JsonResult GetListData(JqGridPostData gridHeader, decimal dossierId, decimal sectionId, bool
firstLoad, decimal dicId)
    {
        var records = waterSituationService.GetListData(dossierId, sectionId, firstLoad, dicId);

        #region сортування
        if (gridHeader.sidx == "Id")
            records = gridHeader.sord == "asc" ? records.OrderBy(x => Convert.ToDecimal(x.Id)).ToList() :
records.OrderByDescending(x => Convert.ToDecimal(x.Id)).ToList();
        else
            records = gridHeader.sord == "asc" ? records.OrderBy(x => x.GRD_NUM1).ToList() :
records.OrderByDescending(x => x.GRD_NUM1).ToList();
        #endregion

        #region пейджинг
        var totalCount = records.Count();

        if (gridHeader.rows != 999999 && totalCount > gridHeader.rows) //если выбрано не "Всі"
            records = records.Skip((gridHeader.page - 1) * gridHeader.rows).Take(gridHeader.rows).ToList();
        else
            gridHeader.page = 1;
        #endregion

        var res = gridHeader.MakeJsonResult(totalCount, records);
        return Json(res, JsonRequestBehavior.AllowGet);
    }

    [ValidateInput(false)]
    public JsonResult EditRow(decimal dossierId, decimal sectionId, decimal dicId, bool onlyEdit, string
jsonObjectString)
    {
        return Json(waterSituationService.EditRow(dossierId, sectionId, dicId, onlyEdit, jsonObjectString),
JsonRequestBehavior.AllowGet);
    }
}

```

**IGDKMapService.cs**

```

public interface IGDKMapService
{
    List<LastGDKData> GetLastData();
}

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						118
Змн.	Арк.	№ докум.	Підпис	Дата		

}

**IMapEcoWaterMonService.cs**

```
public interface IMapEcoWaterMonService
{
    IEnumerable<ListModel> GetDicList();
    Dictionary<string, string> GetChilds(decimal?[] WaterCompName);
    List<EcoWaterMonArea> FilterMap(decimal?[] RiverBasDec, decimal?[] LaboratoryDec, decimal?[]
    WaterCompNameDec, decimal?[] WaterDepNameDec, decimal?[] OPTTypeDec, decimal?[] PeriodOfContDec,
    decimal?[] MonTypeDec, decimal?[] NormTypeDec, decimal?[] TransOPDec);
}
```

**IParsingDocFileService.cs**

```
public interface IParsingDocFileService
{
    /// <summary>
    /// парсинг документа
    /// </summary>
    /// <param name="fileStream">поток файлу</param>
    /// <param name="dossierId">ід екземпляру досьє</param>
    /// <returns></returns>
    string ParseDoc(Stream fileStream, decimal dossierId);
    /// <summary>
    /// Заповнення ґрида
    /// </summary>
    /// <param name="pc">текст документу</param>
    /// <param name="dossierId">ід екземпляру досьє</param>
    /// <returns></returns>
    string FillGrid(ParagraphCollection pc, decimal dossierId);
    /// <summary>
    /// Отримати ід значення словника по локальному коду
    /// </summary>
    /// <param name="code">локальний код DICV_CODE</param>
    /// <param name="dicId">DIC_IDP словника</param>
    /// <returns></returns>
    decimal GetDicIdByDicCode(string code, decimal dicId);
    /// <summary>
    /// Вибірка даних для створення одного запису в ґриді
    /// </summary>
    /// <param name="pc">текст документу</param>
    /// <param name="firstLine">рядок для пошуку</param>
    /// <param name="secondLine">рядок для пошуку</param>
    /// <param name="n">номер рядка, що додається</param>
    /// <returns></returns>
    ModelForGrid SelectData(ParagraphCollection pc, string firstLine, string secondLine, int n);
    /// <summary>
    /// Отримати дату із першої вкладки
    /// </summary>
    /// <param name="dossierId">ід екземпляру досьє </param>
    /// <returns></returns>
    string GetDate(decimal dossierId);
    /// <summary>
    /// Видалення із таблиці даних про водосховища з локальними кодами від 1 до 9
    /// </summary>
    /// <param name="dossierId">ід екзмпляру досьє</param>
    /// <param name="sectionId">ід екземпляру секції</param>
    /// <param name="docCategory">ід категорії документів</param>
}
```

```

/// <param name="saveRowDB">ознак збереження в бд</param>
/// <returns></returns>
string DeleteRecords(decimal dossierId, decimal sectionId, decimal docCategory, bool saveRowDB);
}

```

### IPostService.cs

```

public interface IPostsService
{
    void InsertGridValue(List<TreeModel> arr, decimal dossierId, decimal sectionId);

    List<TreeModel> GetGridData(decimal dossierId, decimal sectionId);
}

```

### IWaterResourcesService.cs

```

public interface IWaterResourcesService
{
    Dossier29003Model ChangeGridAndValues(decimal attr29034, bool editType);
    IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad);

    void DeleteRow(decimal dossierId, decimal sectionId, string date);

    List<string> FillDateArray(decimal dossierId, decimal sectionId);
}

```

### IWaterSituationService.cs

```

public interface IWaterSituationService
{
    /// <summary>
    /// Повертає місяць по локальному коду
    /// </summary>
    /// <param name="dictionaryId">DIC_UID довідника</param>
    /// <param name="localCode">DICV_CODE - локальний код значення довідника</param>
    /// <returns></returns>
    decimal GetMonth(decimal dictionaryId, string localCode);
    /// <summary>
    /// повертає дані для відображення в гріді
    /// </summary>
    /// <param name="dossierId"></param>
    /// <param name="sectionId"></param>
    /// <param name="firstLoad"></param>
    /// <param name="dicId"></param>
    /// <returns></returns>
    IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad, decimal dicId);
    /// <summary>
    /// Додавання/редагування запису в гріді
    /// </summary>
    /// <param name="dossierId"></param>
    /// <param name="sectionId"></param>
    /// <param name="onlyEdit"> true - збереження відредагованого значення в кеш, без розрахунку за формулою/ false - збереження із розрахунком за формулою (використання екстеншена) </param>
    /// <param name="jsonObjectString">доданий/відредагований рядок</param>
    string EditRow(decimal dossierId, decimal sectionId, decimal dicId, bool onlyEdit, string jsonObjectString); }

```

**GDKMapService.cs**

```

public class GDKMapService : IGDKMapService
{
    private readonly IEcoWaterMonRepository repository;

    public GDKMapService(IEcoWaterMonRepository repository)
    {
        this.repository = repository;
    }
    public List<LastGDKData> GetLastData()
    {
        var objects = (from obj in repository.Table<LastDatePostData>()
            select obj).AsQueryable();

        List<LastGDKData> list = new List<LastGDKData>();

        foreach (var item in objects.ToArray())
        {
            LastGDKData data = new LastGDKData
            {
                Amoniy_Excess = item.Amoniy_Excess != null ? item.Amoniy_Excess.ToString() : string.Empty,
                Amoniy_Norm = item.Amoniy_Norm != null ? item.Amoniy_Norm.ToString() : string.Empty,
                Amoniy_Value = item.Amoniy_Value != null ? item.Amoniy_Value.ToString() : string.Empty,
                Azot_Excess = item.Azot_Excess != null ? item.Azot_Excess.ToString() : string.Empty,
                Azot_Norm = item.Azot_Norm != null ? item.Azot_Norm.ToString() : string.Empty,
                Azot_Value = item.Azot_Value != null ? item.Azot_Value.ToString() : string.Empty,
                BSK5_Excess = item.BSK5_Excess != null ? item.BSK5_Excess.ToString() : string.Empty,
                Controle_Date = item.Controle_Date != null ? item.Controle_Date.Value.ToString("dd.MM.yyyy") :
string.Empty,
                Fosfat_Excess = item.Fosfat_Excess != null ? item.Fosfat_Excess.ToString() : string.Empty,
                Hlorid_Excess = item.Hlorid_Excess != null ? item.Hlorid_Excess.ToString() : string.Empty,
                BSK5_Norm = item.BSK5_Norm != null ? item.BSK5_Norm.ToString() : string.Empty,
                BSK5_Value = item.BSK5_Value != null ? item.BSK5_Value.ToString() : string.Empty,
                Fosfat_Norm = item.Fosfat_Norm != null ? item.Fosfat_Norm.ToString() : string.Empty,
                Fosfat_Value = item.Fosfat_Value != null ? item.Fosfat_Value.ToString() : string.Empty,
                Hlorid_Norm = item.Hlorid_Norm != null ? item.Hlorid_Norm.ToString() : string.Empty,
                Hlorid_Value = item.Hlorid_Value != null ? item.Hlorid_Value.ToString() : string.Empty,
                Kisen_Excess = item.Kisen_Excess != null ? item.Kisen_Excess.ToString() : string.Empty,
                Kisen_Norm = item.Kisen_Norm != null ? item.Kisen_Norm.ToString() : string.Empty,
                Kisen_Value = item.Kisen_Value != null ? item.Kisen_Value.ToString() : string.Empty,
                Latitude = item.Latitude != null ? item.Latitude.ToString() : string.Empty,
                Longitude = item.Longitude != null ? item.Longitude.ToString() : string.Empty,
                Nitrat_Excess = item.Nitrat_Excess != null ? item.Nitrat_Excess.ToString() : string.Empty,
                Nitrat_Norm = item.Nitrat_Norm != null ? item.Nitrat_Norm.ToString() : string.Empty,
                Nitrat_Value = item.Nitrat_Value != null ? item.Nitrat_Value.ToString() : string.Empty,
                Nitrit_Excess = item.Nitrit_Excess != null ? item.Nitrit_Excess.ToString() : string.Empty,
                Nitrit_Norm = item.Nitrit_Norm != null ? item.Nitrit_Norm.ToString() : string.Empty,
                Nitrit_Value = item.Nitrit_Value != null ? item.Nitrit_Value.ToString() : string.Empty,
                Post_Name = item.Post_Name != null ? item.Post_Name.ToString() : string.Empty,
                Riverbas_Name = item.Riverbas_Name != null ? item.Riverbas_Name.ToString() : string.Empty,
                Sulfat_Excess = item.Sulfat_Excess != null ? item.Sulfat_Excess.ToString() : string.Empty,
                Sulfat_Norm = item.Sulfat_Norm != null ? item.Sulfat_Norm.ToString() : string.Empty,
                Sulfat_Value = item.Sulfat_Value != null ? item.Sulfat_Value.ToString() : string.Empty,
                Total_Index = item.Total_Index != null ? item.Total_Index.ToString() : string.Empty,
                WaterLab_Name = item.WaterLab_Name != null ? item.WaterLab_Name.ToString() : string.Empty,
                Zavisli_Excess = item.Zavisli_Excess != null ? item.Zavisli_Excess.ToString() : string.Empty,
                Zavisli_Norm = item.Zavisli_Norm != null ? item.Zavisli_Norm.ToString() : string.Empty,
                Zavisli_Value = item.Zavisli_Value != null ? item.Zavisli_Value.ToString() : string.Empty,
                IDCard_Monitoring = item.IDCard_Monitoring,
                Report_ID = item.Report_ID
            }
        }
    }
}

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						121
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    };
    list.Add(data);
}
return list;
}
}

```

### MapEcoWaterMonService.cs

```

public class MapEcoWaterMonService : IMapEcoWaterMonService
{
    private readonly IEcoWaterMonRepository repository;

    public MapEcoWaterMonService(IEcoWaterMonRepository repository)
    {
        this.repository = repository;
    }

    public IEnumerable<ListModel> GetDicList()
    {
        IEnumerable<ListModel> result = (from dv in repository.Table<Dictionary_Value>()
                                         where (dv.DIC_IDP == 29001 || dv.DIC_IDP == 29003 || dv.DIC_IDP == 29005 ||
dv.DIC_IDP == 29007 ||
                                         dv.DIC_IDP == 29023 || dv.DIC_IDP == 29025 || dv.DIC_IDP == 29021 || dv.DIC_IDP
== 29019 || dv.DIC_IDP == 29027)
                                         select new ListModel
                                         {
                                             id = dv.DIC_IDP,
                                             Text = dv.DICV_LNAM,
                                             Value = Convert.ToString(dv.DICV_UID)
                                         }).OrderBy(x => x.Text).ToList();

        return result;
    }

    public Dictionary<string, string> GetChilds(decimal?[] WaterCompName)
    {
        Dictionary<string, string> d = new Dictionary<string, string>();
        var objects = (from obj in repository.Table<Dictionary_Value>()
                       where (obj.DIC_IDP == 29007)
                       select obj).AsQueryable();
        /*if (WaterCompType[0] != null)*/ objects = objects.Where(x =>
WaterCompName.Contains(Convert.ToDecimal(x.DICV_LID)));

        foreach (var o in objects)
        {
            d.Add(o.DICV_IDP.ToString(), o.DICV_LNAM);
        }

        return d;
    }

    public List<EcoWaterMonArea> FilterMap(decimal?[] RiverBasDec, decimal?[] LaboratoryDec, decimal?[]
WaterCompNameDec, decimal?[] WaterDepNameDec, decimal?[] OPTYPEDec, decimal?[] PeriodOfContDec,
decimal?[] MonTypeDec, decimal?[] NormTypeDec, decimal?[] TransOPDec)
    {
        var objects = (from obj in repository.Table<select_all_OP>()
                       select obj).AsQueryable();

        if (RiverBasDec[0] != null) objects = objects.Where(x =>

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						122
Змн.	Арк.	№ докум.	Підпис	Дата		

```

RiverBasDec.Contains(Convert.ToDecimal(x.RiverBasID)));
    if (LaboratoryDec[0] != null) objects = objects.Where(x =>
LaboratoryDec.Contains(Convert.ToDecimal(x.LaboratoryID)));
    if (WaterCompNameDec[0] != null) objects = objects.Where(x =>
WaterCompNameDec.Contains(Convert.ToDecimal(x.WaterCompNameID)));
    if (WaterDepNameDec[0] != null) objects = objects.Where(x =>
WaterDepNameDec.Contains(Convert.ToDecimal(x.WaterDepNameID)));
    if (OPTypeDec[0] != null) objects = objects.Where(x =>
OPTypeDec.Contains(Convert.ToDecimal(x.OPTypeID)));
    if (PeriodOfContDec[0] != null) objects = objects.Where(x =>
PeriodOfContDec.Contains(Convert.ToDecimal(x.PeriodOfContID)));
    if (MonTypeDec[0] != null) objects = objects.Where(x =>
MonTypeDec.Contains(Convert.ToDecimal(x.MonTypeID)));
    if (NormTypeDec[0] != null) objects = objects.Where(x =>
NormTypeDec.Contains(Convert.ToDecimal(x.NormTypeID)));
    if (TransOPDec[0] != null) objects = objects.Where(x =>
TransOPDec.Contains(Convert.ToDecimal(x.TransOPID)));

EcoWaterMonArea currentArea = null;
EcoWaterMonArea previousCurrentArea = null;
EcoWaterMonAreaInfo areaInfo = null;
EcoWaterMonAreaInfo previousAreaInfo = null;
bool addCords = true;
var areas = new List<EcoWaterMonArea>();
foreach (var item in objects.ToArray())
{
    addCords = true;
    //якщо нова Area, то створюємо її, то змінюємо currentArea
    if (currentArea == null || item.ObjectID != areaInfo.Id)
    {
        areaInfo = new EcoWaterMonAreaInfo
        {
            Id = (int)item.ObjectID, //id об'єкта
            Name = item.Name, //назва об'єкта

            RiverBas = item.RiverBas != null ? item.RiverBas : string.Empty,
            Laboratory = item.Laboratory != null ? item.Laboratory : string.Empty,
            MonType = item.MonType != null ? item.MonType : string.Empty,
            NormType = item.NormType != null ? item.NormType : string.Empty,
            OPCode = item.OPCode != null ? item.OPCode : string.Empty,
            OPStatus = item.OPStatus != null ? item.OPStatus : string.Empty,
            OPType = item.OPType != null ? item.OPType : string.Empty,
            PeriodOfCont = item.PeriodOfCont != null ? item.PeriodOfCont : string.Empty,
            TransOP = item.TransOP != null ? item.TransOP : string.Empty,
            WaterCompName = item.WaterCompName != null ? item.WaterCompName : string.Empty,
            WaterDepName = item.WaterDepName != null ? item.WaterDepName : string.Empty,
            ReportId = item.ReportID,
            DossierId = item.DossierID
        };

        currentArea = new EcoWaterMonArea
        {
            Coords = new List<EcoWaterMonPoint>(),
            ChildObjects = new List<EcoWaterMonAreaInfo>(),
        };

        areas.Add(currentArea);
        currentArea.ChildObjects.Add(areaInfo);
        previousAreaInfo = areaInfo;
        previousCurrentArea = currentArea;
    } if (addCords)

```



```

{
    // додаємо координати точки до поточної area
    currentArea.Coords.Add(new EcoWaterMonPoint
    {
        lat = decimal.Parse(item.Latitude.Replace(".", ",")),
        lng = decimal.Parse(item.Longitude.Replace(".", ","))
    });
}
}
return areas;
}
}

```

### ParsingDocFileService.cs

```

public class ParsingDocFileService : IParsingDocFileService
{
    private IDossierItemManager _dossierItemManager;
    private IDossierAttributeManager _dossierAttributeManager;
    private ITableService _tableService;
    private IDocsService _docService;
    private IDossierRepository _repository;

    public ParsingDocFileService(IDossierItemManager dossierItemManager, IDossierAttributeManager dossierAttributeManager, ITableService tableService, IDossierRepository repository, IDocsService docService)
    {
        _dossierAttributeManager = dossierAttributeManager;
        _dossierItemManager = dossierItemManager;
        _repository = repository;
        _docService = docService;
        _tableService = tableService;
    }

    public string ParseDoc(Stream fileStream, decimal dossierId)
    {
        fileStream.Position = 0;
        Document doc = new Document(fileStream, FileFormat.Doc);
        ParagraphCollection pc = doc.Sections[0].Paragraphs;

        return FillGrid(pc, dossierId);
    }

    public string FillGrid(ParagraphCollection pc, decimal dossierId)
    {
        string date = String.Format("станом на {0}", GetDate(dossierId));
        bool isExist = false;
        List<DossierAttribute> records = new List<DossierAttribute>();
        var dossier = _dossierItemManager.GetDossierItem(dossierId);

        try
        {
            for (int i = 1; i < 10; i++)
            {
                DossierAttribute data = _docService.CreateDoc(29014, true).FindAttribute(29096);
                var model = SelectData(pc, "Гідрологічна ситуація на водосховищах України", date, i);
                if (model.ActualDiscount == 0 && model.ActualInflux == 0 && model.ActualLevel == 0)
                {
                    return "1";//string.Format(EcoWaterMonResources.DoesntExistsDate);
                }
                data.Value = 0;
                data.Attributes["29097"].Value = model.ReservoirName;
            }
        }
        catch { }
    }
}

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						124
Змн.	Арк.	№ докум.	Підпис	Дата		

```

data.Attributes["29098"].Value = "";
data.Attributes["29099"].Value = "";
data.Attributes["29100"].Value = model.ActualLevel;
data.Attributes["29101"].Value = "";
data.Attributes["29102"].Value = model.ActualInflux;
data.Attributes["29103"].Value = model.ActualDiscount;
data.Attributes["29104"].Value = "";
data.Attributes["29105"].Value = "";

records.Add(data);
var existItem = IsExistReservoir(dossierId, data);
if(existItem)
{
    isExist = true;
    break;
}
}
if (isExist)
{
    return "2";//EcoWaterMonResources.DataExistInTable;
}

else
{
    for (int i = 0; i < records.Count; i++)
    {
        _tableService.EditGridItem(dossierId, dossier.Sections["29015"].ItemId, records[i], null, 0, 1);
    }
}

}
catch(Exception e)
{
    return "3";// string.Format(EcoWaterMonResources.ErrorFormat);
}

return "4";
}
public ModelForGrid SelectData(ParagraphCollection pc, string firstLine, string secondLine, int n)
{
    ModelForGrid model = new ModelForGrid();

    model.ReservoirName = GetDicIdByDicCode(n.ToString(), 29049);
    string[] columns = new string[20];
    for (int i = 0; i < pc.Count - 1; i++)
    {
        if (pc[i].Text.Trim() == firstLine && pc[i + 1].Text.Trim() == secondLine)
        {
            if (n == 9)
            {
                n++;
                columns = pc[i + 10 + n].Text.Split(':');
                break;
            }
        }
    }
    if (columns[0] != null)
    {
        if (n == 1)
        {

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						125
Змн.	Арк.	№ докум.	Підпис	Дата		

```

if (columns[14].Trim() != "")
    model.ActualInflux = Convert.ToDecimal(columns[14].Trim());
if (columns[15].Trim() != "")
    model.ActualDiscount = Convert.ToDecimal(columns[15].Trim());

for (int i = 0; i < pc.Count - 1; i++)
{
    if (pc[i].Text.Trim() == "Гідрологічна ситуація на водпостах Київ, Канів" && pc[i +
1].Text.Trim() == secondLine)
    {
        var col = pc[i + 5].Text.Split(':');
        if (col[5].Trim() != "")
            model.ActualLevel = (Convert.ToDecimal(col[5].Trim()) + 10000)/100;
        break;
    }
}

if (n == 3 || n == 6)
{
    if (columns[4].Trim() != "")
        model.ActualLevel = Convert.ToDecimal(columns[4].Trim())/100;
}

if (n == 2 || n == 4 || n == 5 || n == 7 || n == 8)
{
    if (columns[3].Trim() != "")
        model.ActualLevel = Convert.ToDecimal(columns[3].Trim().Replace(".", ","))/100;
}

if (n == 10)
{
    if (columns[2].Trim() != "")
        model.ActualLevel = Convert.ToDecimal(columns[2].Trim())/100;
}

if (columns[14].Trim() != "")
    model.ActualInflux = Convert.ToDecimal(columns[14].Trim().Replace(".", ","));
if (columns[15].Trim() != "")
    model.ActualDiscount = Convert.ToDecimal(columns[15].Trim().Replace(".", ","));
}

return model;
}

public string DeleteRecords(decimal dossierId, decimal sectionId, decimal docCategory, bool saveRowDB)
{
    var dossier = _dossierItemManager.GetDossierItem(dossierId);
    for (int i = 1; i < 10; i++)
    {
        var code = GetDicIdByDicCode(i.ToString(), 29049);
        var recordId =
        ((DataTable)dossier.Sections["29015"].Attributes["29096"].Value).Select(string.Format("[29097] =
'{0}'","",int.Parse(code.ToString()))).FirstOrDefault();
        if (recordId != null)
            _tableService.DeleteItem(dossierId, sectionId, 29096, Convert.ToDecimal(recordId["Id"]), docCategory,
saveRowDB);
    }
    return "";
}

```

```

public decimal GetDicIdByDicCode(string code, decimal dicId)
{
    decimal val = (from dv in _repository.Table<Dictionary_Value>()
        where dv.DICV_CODE == code && dv.DIC_IDP == dicId
        select dv.DICV_UID).FirstOrDefault();
    return val;
}

public string GetDate(decimal dossierId)
{
    var dossier = _dossierItemManager.GetDossierItem(dossierId);
    return
    Convert.ToDateTime(dossier.Sections["29015"].Attributes["29093"].Value).ToString("dd.MM.yyyy");
}

private bool IsExistReservoir(decimal dossierId, DossierAttribute newAttributes)
{
    var dossier = _dossierItemManager.GetDossierItem(dossierId);
    var grid = ((DataTable)dossier.Sections["29015"].Attributes["29096"].Value).
        Select(string.Format("[29097] = '{0}' and [Id] <> '{1}'",
            int.Parse(newAttributes.Attributes["29097"].Value.ToString()),
            newAttributes.Value
        ));

    if (grid != null && grid.Count() != 0)
    {
        return true;
    }
    return false;
}
}

```

### WaterResourcesService.cs

```

public class WaterResourcesService : IWaterResourcesService
{
    private IDossierRepository repository;
    private IDossierAttributeManager dossierAttributeManager;
    private IDossierItemManager dossierItemManager;

    public WaterResourcesService(IDossierRepository repository, IDossierAttributeManager
    dossierAttributeManager, IDossierItemManager dossierItemManager)
    {
        this.repository = repository;
        this.dossierAttributeManager = dossierAttributeManager;
        this.dossierItemManager = dossierItemManager;
    }

    public Dossier29003Model ChangeGridAndValues(decimal attr29034, bool editType)
    {
        var dossier = dossierItemManager.GetDossierItem(attr29034, true);
        Dossier29003Model model = new Dossier29003Model()
        {
            attr29035 = dossier.Sections["29001"].Attributes["29006"].Value != null ?
            Convert.ToDecimal(dossier.Sections["29001"].Attributes["29006"].Value) : 0,
            attr29063 = dossier.Sections["29001"].Attributes["29005"].Value != null ?
            Convert.ToDecimal(dossier.Sections["29001"].Attributes["29005"].Value) : 0,
            attr29036 = dossier.Sections["29001"].Attributes["29008"].Value != null ?
            Convert.ToDecimal(dossier.Sections["29001"].Attributes["29008"].Value) : 0,
            attr29037 = dossier.Sections["29001"].Attributes["29009"].Value != null ?

```

```

Convert.ToDecimal(dossier.Sections["29001"].Attributes["29009"].Value) : 0,
    attr29039 = dossier.Sections["29001"].Attributes["29016"].Value != null ?
dossier.Sections["29001"].Attributes["29016"].Value.ToString() : string.Empty,
    attr29038 = dossier.Sections["29001"].Attributes["29018"].Value != null ?
dossier.Sections["29001"].Attributes["29018"].Value.ToString() : string.Empty,
    attr29041 = dossier.Sections["29001"].Attributes["29020"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29020"].Value) : 0,
    attr29042 = dossier.Sections["29001"].Attributes["29021"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29021"].Value) : 0,
    attr29040 = dossier.Sections["29001"].Attributes["29022"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29022"].Value) : 0,
    attr29043 = dossier.Sections["29001"].Attributes["29023"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29023"].Value) : 0,
    attr29044 = dossier.Sections["29001"].Attributes["29024"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29024"].Value) : 0,
    attr29045 = dossier.Sections["29001"].Attributes["29025"].Value != null ?
Convert.ToDecimal(dossier.Sections["29001"].Attributes["29025"].Value) : 0,
    colModel = new List<Grid29048Model>(),
    groupHeaders = new List<GroupHeadersModel>(),
    colNames = new List<string>()
};

List<GridDataModel> fieldValues = new
JavaScriptSerializer().Deserialize<List<GridDataModel>>(dossier.Sections["29001"].Attributes["29032"].Value.To
String());

model.colModel.Add(new Grid29048Model()
{
    hidden = true,
    label = string.Empty,
    name = "Id",
    sortable = false,
});

if (editType)
{
    model.colModel.Add(new Grid29048Model()
    {
        name = "delete",
        label = string.Empty,
        index = "",
        sortable = false,
        width = 22
    });

    model.colModel.Add(new Grid29048Model()
    {
        name = "edit",
        label = string.Empty,
        index = "",
        sortable = false,
        width = 22,
    });
}
model.colModel.Add(new Grid29048Model()
{
    hidden = false,
    align = "left",
    classes = "height-formatter",
    index = "GRD_DATE1",
    label = "Дата відбору проби",

```

Змн.	Арк.	№ докум.	Підпис	Дата

```
name = "GRD_DATE1",
resizable = false,
sortable = true,
editable = true,
width = 100,
});
foreach (var item in fieldValues)
{
    model.colModel.Add(new Grid29048Model()
    {
        hidden = false,
        align = "left",
        classes = "height-formatter",
        index = item.indicatorId,
        label = item.indicatorName,
        name = item.indicatorId,
        resizable = false,
        sortable = false,
        editable = true,
        width = 100
    });

    if (model.groupHeaders.Where(x => x.titleText == item.parentId).Count() == 0)
    {
        model.groupHeaders.Add(new GroupHeadersModel()
        {
            startColumnName = item.indicatorId,
            numberOfColumns = fieldValues.Where(x => x.parentId == item.parentId).Count(),
            titleText = item.parentId
        });
    }
}

foreach (var name in model.colModel)
{
    model.colNames.Add(name.label);
}

return model;
}

public IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad)
{
    List<ListDataModel> result = new List<ListDataModel>();

    var grdAttr = (DataTable)dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29048, firstLoad,
29003).Value;

    foreach (DataRow row in grdAttr.Rows)
    {
        ListDataModel model = new ListDataModel { GRD_DATE1 = null, GRD_NUM1 = null, GRD_NUM2 =
null, GRD_NUM3 = null };
        model.GRD_DATE1 = Convert.ToDateTime(row["29049"].ToString());
        model.GRD_NUM1 = Convert.ToDecimal(row["29050"]);
        model.GRD_NUM2 = Convert.ToDecimal(row["29051"]);
        if (row["29052"].ToString() != string.Empty)
            model.GRD_NUM3 = Convert.ToDecimal(row["29052"]);
        result.Add(model);
    }
    return result;
}
```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						129
Змн.	Арк.	№ докум.	Підпис	Дата		

```

public void DeleteRow(decimal dossierId, decimal sectionId, string date)
{
    var grdAttr = (DataTable)dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29048, false,
0).Value;
    DateTime dt = Convert.ToDateTime(date);
    var existingRows = grdAttr.AsEnumerable().Where(x => x.Field<DateTime>("29049") == dt);

    List<DataRow> deleteList = new List<DataRow>();
    foreach (var item in existingRows)
    {
        deleteList.Add(item);
    }

    foreach (var item in deleteList)
    {
        grdAttr.Rows.Remove(item);
    }
}

public List<string> FillDateArray(decimal dossierId, decimal sectionId)
{
    var grdAttr = (DataTable)dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29048, true, 0).Value;
    List<string> dateList = new List<string>();

    foreach (DataRow row in grdAttr.Rows)
    {
        dateList.Add(Convert.ToDateTime(row["29049"]).ToString("dd.MM.yyyy"));
    }

    return dateList.Distinct().ToList();
} }

```

### WaterSituationService.cs

```

public class WaterSituationService : IWaterSituationService
{
    private IDictionaryService dictionaryService;
    private IDocsService docService;
    private ITableService tableService;
    private IDossierAttributeManager dossierAttributeManager;
    private IDictionaryValuesCacheManager dictionaryManager;
    private IDossierItemManager dossierItemManager;

    public WaterSituationService(IDictionaryService dictionaryService, IDocsService docService,
IDossierAttributeManager dossierAttributeManager, IDictionaryValuesCacheManager dictionaryManager,
IDossierItemManager dossierItemManager, ITableService tableService)
    {
        this.dictionaryService = dictionaryService;
        this.docService = docService;
        this.tableService = tableService;
        this.dossierAttributeManager = dossierAttributeManager;
        this.dictionaryManager = dictionaryManager;
        this.dossierItemManager = dossierItemManager;
    }
    public decimal GetMonth(decimal dictionaryId, string localCode)
    {
        var dictionary = dictionaryService.LoadDictionary(dictionaryId);
        var dicAttr = (from item in dictionary.Attributes
            where item.Id == 29055
            select item).FirstOrDefault();
        var valueId = (from val in dicAttr.Values

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						130
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        where val.LocalCode == localCode
        select val.Id).FirstOrDefault();
    return valueId;
}

public IEnumerable<ListDataModel> GetListData(decimal dossierId, decimal sectionId, bool firstLoad,
decimal dicId)
{
    List<ListDataModel> result = new List<ListDataModel>();
    List<decimal> recordFromDB = new List<decimal>();

    var grdAttr = (DataTable)dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29106, firstLoad,
29016).Value;

    // обираємо записи із бд
    foreach (DataRow row in grdAttr.Rows)
    {
        ListDataModel model = new ListDataModel();
        model.Id = Convert.ToDecimal(row["Id"]);
        model.GRD_NUM1 = dictionaryManager.GetValue(dicId, Convert.ToDecimal(row["29107"])).Name;
        if (row["29108"].ToString() != string.Empty)
            model.GRD_NUM2 = Convert.ToDecimal(row["29108"]);
        if (row["29109"].ToString() != string.Empty)
            model.GRD_NUM3 = Convert.ToDecimal(row["29109"]);
        if (row["29110"].ToString() != string.Empty)
            model.GRD_NUM4 = Convert.ToDecimal(row["29110"]);
        model.GRD_NUM5 = Convert.ToDecimal(row["29111"]);

        result.Add(model);
        recordFromDB.Add(Convert.ToDecimal(row["29107"]));
    }

    // отримуємо всі значення словника "Найменування каналу"
    var dicValues = dictionaryManager.GetValues(dicId).Where(x => x.DateEnd > DateTime.Now);

    // дописуємо рядки, що лишилися (по значенням довідника)
    decimal i = -10000;
    foreach( var val in dicValues)
    {
        var isExist = recordFromDB.Contains(val.Id);
        if(!isExist)
        {
            i--;
            ListDataModel model = new ListDataModel();
            model.Id = i;
            model.GRD_NUM1 = dictionaryManager.GetValue(dicId, Convert.ToDecimal(val.Id)).Name;
            result.Add(model);
        }
    }
    return result;
}

public string EditRow(decimal dossierId, decimal sectionId, decimal dicId, bool onlyEdit, string
jsonObjectString)
{
    try
    {
        //формируем объект с данными который пришел из клиента
        Dictionary<string, string> values = new JavaScriptSerializer().Deserialize<Dictionary<string,
string>>(jsonObjectString);
        var grdAttr = (DataTable)dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29106, false,

```

					ДП ІС-5109.1181-с.ПЗ	Арк.
						131
Змн.	Арк.	№ докум.	Підпис	Дата		



```

29014).Value;
List<DossierAttribute> list = new List<DossierAttribute>();

var dossier = dossierItemManager.GetDossierItem(dossierId);
decimal id = Convert.ToDecimal(values["Id"]);
var waterObject = values.Where(x => x.Key == "GRD_NUM1").First().Value;

//выбираем из списка записей строку для сохранения изменений
var isExistRow = grdAttr.AsEnumerable().Where(x => x.Field<Decimal>("Id") == id).FirstOrDefault();
DossierAttribute data = docService.CreateDoc(29014).FindAttribute(29106);

var indicator = values.Where(x => x.Key == "GRD_NUM5").First().Value;
// если редактируется ранее добавленная строка
if (isExistRow != null && (id > 0 || id > -1000) && indicator != "")
{
    data.Value = isExistRow["Id"];
    //data.ParentDossierId = 0;
    data.Attributes["29107"].Value = dictionaryManager.GetDictionaryAttribute(dicId).Values.Where(x =>
x.Name == waterObject).First().Id;
    var temp1 = values.Where(x => x.Key == "GRD_NUM2").First().Value;
    data.Attributes["29108"].Value = temp1 != "" ? Convert.ToDouble(temp1.Replace(".", ",")).ToString() :
"";

    var temp2 = values.Where(x => x.Key == "GRD_NUM3").First().Value;
    data.Attributes["29109"].Value = temp2 != "" ? Convert.ToDouble(temp2.Replace(".", ",")).ToString() :
"";

    var temp3 = values.Where(x => x.Key == "GRD_NUM4").First().Value;
    data.Attributes["29110"].Value = temp3 != "" ? Convert.ToDouble(temp3.Replace(".", ",")).ToString() :
"";

    var temp4 = values.Where(x => x.Key == "GRD_NUM5").First().Value;
    data.Attributes["29111"].Value = temp4 != "" ? Convert.ToDouble(temp4.Replace(".", ",")).ToString() :
"";

    if (data != null)
    {
        if (onlyEdit)
        {
            var table = dossierAttributeManager.LoadAttribute(dossierId, sectionId, 29106, false, 29014);
            tableService.AddGridItem(data, table); // сохраняем в памяти без расчета по формулам
        }
        else // сохраняем с расчетом по формулам
            tableService.EditGridItem(dossierId, Convert.ToDecimal(dossier.Sections[1].ItemId), data, null, 0,
0);
    }
}

// если id<0 значит это строка, которую только добавили
else if (id < -1000 && indicator != "")
{
    data.Value = 0;
    // data.ParentDossierId = 0;
    data.Attributes["29107"].Value = dictionaryManager.GetDictionaryAttribute(dicId).Values.Where(x =>
x.Name == waterObject).First().Id;
    data.Attributes["29108"].Value = DBNull.Value;
    data.Attributes["29109"].Value = DBNull.Value;
    data.Attributes["29110"].Value = DBNull.Value;
    data.Attributes["29111"].Value = Convert.ToDouble(indicator.ToString().Replace(".", ",")).ToString();

    if (data != null)
        tableService.EditGridItem(dossierId, Convert.ToDecimal(dossier.Sections[1].ItemId), data, null, 0,
0);
}

```

					ДП-1811.6019-СИ ПП	Арк.
						132
Змн.	Арк.	№ докум.	Підпис	Дата		

```
else // удаление
{
    DataRow row = grdAttr.Rows.Find(id);
    grdAttr.Rows.Remove(row);
}
}
catch(Exception ex)
{
    return ex.Message;
}
return String.Empty;
}
}
```



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”  
Кафедра автоматизованих систем обробки інформації та управління

**УЗГОДЖЕНО**

**Керівник проекту**

\_\_\_\_\_  
(підпис) М.О. Сперкач  
(ініціали, прізвище)

“16” квітня 2019 р.

**ЗАТВЕРДЖУЮ**

**Начальник відділу адміністрування  
водного кадастру та моніторингу  
вод**

\_\_\_\_\_  
(підпис) М.М. Шпанчик  
(ініціали, прізвище)

“17” квітня 2019 р.

Інформаційна система моніторингу та екологічної оцінки водних  
ресурсів України

**ТЕХНІЧНЕ ЗАВДАННЯ**

Шифр ДП ІС-5109.1181-с.ТЗ

на 16 сторінках

Київ – 2019 року

## ЗМІСТ

1	ЗАГАЛЬНІ ПОЛОЖЕННЯ .....	3
1.1	Повне найменування системи та її умовне позначення.....	3
1.2	Найменування організації-замовника та учасників робіт .....	3
1.3	Перелік документів, на підставі яких створюється система .....	3
1.4	Планові терміни початку і закінчення роботи зі створення системи .	4
2	ПРИЗНАЧЕННЯ І МЕТА СТВОРЕННЯ КОМПЛЕКСУ ЗАДАЧ.....	5
2.1	ПРИЗНАЧЕННЯ КОМПЛЕКСУ ЗАДАЧ .....	5
2.2	ЦІЛІ СТВОРЕННЯ КОМПЛЕКСУ ЗАДАЧ .....	5
3	ХАРАКТЕРИСТИКА ОБ’ЄКТА АВТОМАТИЗАЦІЇ.....	7
4	ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	8
4.1	Вимоги до функціональних характеристик.....	8
4.2	Вимоги до надійності .....	13
4.3	Вимоги до складу і параметрів технічних засобів.....	13
5	СТАДІЇ І ЕТАПИ РОЗРОБКИ.....	15
6	ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ СИСТЕМИ.....	16
6.1	Види випробувань.....	16

					ДП ІС-5109.1181-с.ТЗ			
		Прізвище	Підпис	Дата				
Розроб.		Казмірчук А.В.			Інформаційна система моніторингу та екологічної оцінки водних ресурсів України	Лім.	Лист	Листів
Перевірів		Сперкач М.О.					2	16
Н. кон.		Тєлишева Т.О.				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51		
Затв.		Павлов О.А.						

## 1 ЗАГАЛЬНІ ПОЛОЖЕННЯ

### 1.1 Повне найменування системи та її умовне позначення

Повне найменування системи: «Інформаційна система моніторингу та екологічної оцінки водних ресурсів України».

### 1.2 Найменування організації-замовника та організацій-учасників робіт

Генеральним замовником проекту є Державне агенство водних ресурсів України. Адреса замовника: м.Київ, вул. Велика Васильківська, 8.

Розробником системи є студентка групи ІС-51 кафедри Автоматизованих систем обробки інформації та управління Національного технічного університету України "Київський політехнічний інститут імені Ігоря Сікорського" Казмірчук Аліна Василівна.

### 1.3 Перелік документів, на підставі яких створюється система

При розробці системи і створення проектно-експлуатаційної документації Виконавець повинен керуватися вимогами наступних нормативних документів:

- ДСТУ 19.201-78. Технічне завдання. Вимоги до змісту і оформлення;
- ДСТУ 34.601-90. Комплекс стандартів на автоматизовані системи. Автоматизовані системи. Стадії створення;
- ДСТУ 34.201-89. Інформаційні технології. Комплекс стандартів на автоматизовані системи. Види, комплексність і позначення документів при створенні автоматизованих систем.

					ДП ІС-5109.1181-с.ТЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

#### 1.4 Планові терміни початку і закінчення роботи зі створення системи

Плановий строк початку роботи по створенню інформаційної системи моніторингу та екологічної оцінки водних ресурсів України 15 квітня 2019 року.

Плановий строк кінця роботи по створенню інформаційної системи моніторингу та екологічної оцінки водних ресурсів України – не пізніше 1 червня 2019 року.

					ДП ІС-5109.1181-с.ТЗ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

## 2 ПРИЗНАЧЕННЯ І МЕТА СТВОРЕННЯ КОМПЛЕКСУ ЗАДАЧ

### 2.1 Призначення комплексу задач

Призначенням розробки є підтримка процесу проведення моніторингу та екологічної оцінки водних ресурсів України.

### 2.2 Цілі створення комплексу задач

Основними цілями створення інформаційної системи моніторингу та екологічної оцінки водних ресурсів України є:

- покращення інформаційного супроводу робіт із здійснення моніторингу поверхневих вод та екологічної оцінки водних ресурсів України;
- підвищення оперативності та ефективності аналізу наявної інформації про стан водних ресурсів;
- спрощення регулярної актуалізації і багатоцільового використання інформації стосовно стану водних ресурсів.

Для досягнення поставлених цілей мають бути вирішені такі задачі:

- реєстрація пунктів спостереження;
- ведення моніторингу пунктів спостереження;
- можливість систематизації результатів довгострокових спостережень за станом водних ресурсів, його кількісними та якісними показниками;
- ведення контролю водогосподарської обстановки;
- створення карти перевищення ГДК показників на постах спостереження;
- створення карти розміщення пунктів спостереження;
- заповнення довідників;
- заповнення ГДК;
- внесення нормативів;



- формування звітності доступного типу за певний період;
- керування користувачами системи.

					ДП ІС-5109.1181-с.ТЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3 ХАРАКТЕРИСТИКА ОБ'ЄКТА АВТОМАТИЗАЦІЇ

Об'єктом автоматизації є Держводагенство, де необхідно автоматизувати (комп'ютеризувати) процеси ведення моніторингу постів спостереження та даних водогосподарської обстановки, обробку отриманих даних для підвищення оперативності та ефективності аналізу стану водних ресурсів.

Результатом цього буде покращення інформаційного супроводу моніторингу поверхневих вод України, зменшення часу аналізу та формування звітності для подальшого прийняття рішень. Всі дані моніторингу будуть зберігатися в єдиному форматі для всіх водогосподарських організацій в єдиній базі даних.

					ДП ІС-5109.1181-с.ТЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

## 4 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 4.1 Вимоги до функціональних характеристик

Дана розробка повинна забезпечувати підтримку процесу проведення моніторингу та оцінки водних ресурсів України. Система має виконувати наступні функції:

1. система повинна надавати можливість реєстрації поста спостереження;

1.1. система повинна надавати можливість створення карточки поста спостереження;

1.1.1. система повинна надавати можливість введення всіх необхідних атрибутів поста спостереження;

1.1.2. система повинна надавати можливість визначення координат поста спостереження для відображення на карті;

1.1.3. система повинна надавати можливість вибору показників для моніторингу певного поста спостереження;

1.2. система повинна надавати можливість редагування карточки поста спостереження;

1.3. система повинна надавати можливість видалення карточки поста спостереження;

2. система повинна надавати можливість ведення моніторингу поста спостереження;

2.1. система повинна надавати можливість створення карточки моніторингу поста спостереження;

2.1.1. система повинна надавати можливість вибору поста спостереження;

2.1.2. система повинна надавати можливість внесення даних до таблиці спостережень за конкретну дату спостереження;

- 2.1.3. система повинна надавати можливість редагування даних таблиці спостережень за конкретну дату спостережень;
- 2.1.4. система повинна надавати можливість видалення даних з таблиці спостережень за конкретну дату спостережень;
- 2.2. система повинна надавати можливість редагування карточки моніторингу поста спостереження;
- 2.3. система повинна надавати можливість видалення карточки моніторингу поста спостереження;
- 3. система повинна надавати можливість ведення нормативів ГДК;
  - 3.1. система повинна надавати можливість створення карточки нормативів ГДК для показника;
    - 3.1.1. система повинна надавати можливість вибору групи показників;
    - 3.1.2. система повинна надавати можливість вибору показника;
    - 3.1.3. система повинна надавати можливість додавання нормативів показника з мінімальним та максимальним допустимим значенням;
    - 3.1.4. система повинна надавати можливість редагування/видалення нормативів показника;
  - 3.2. система повинна надавати можливість редагування карточки нормативу ГДК;
  - 3.3. система повинна надавати можливість видалення карточки нормативу ГДК;
- 4. система повинна надавати можливість перегляду карти перевищення ГДК показників на постах спостереження;
  - 4.1. система повинна відображати зареєстровані пости спостереження на карті;

4.1.1. система повинна відображати пости спостереження маркерами, колір і розмір яких відповідає рівню і кількості перевищень ГДК показників;

4.1.2. система повинна при наведенні на маркер надавати коротку реєстраційну інформацію про пост спостереження і значення показників на останню дату спостережень;

4.1.3. система повинна при виборі поста спостереження на карті надавати можливість переходу на формування звіту по даних моніторингу даного поста спостереження;

5. система повинна надавати можливість перегляду карти розміщення постів спостереження;

5.1. система повинна відображати зареєстровані пости спостереження на карті;

5.1.1. система повинна при наведенні на відповідний пост спостереження, що розміщений на карті, надавати коротку реєстраційну інформацію про пост спостереження;

5.1.2. система повинна при виборі поста спостереження на карті надавати можливість переходу на формування звіту по даних моніторингу даного поста спостереження;

6. система повинна надавати можливість ведення нормативів для контролю водогосподарської обстановки;

6.1. система повинна надавати можливість створення карточки нормативу координат кривих водосховища;

6.1.1. система повинна надавати можливість вибору основного басейна, водосховища, внесення проектного рівня та проектного обсягу, координат кривих об'ємів водосховища;

6.2. система повинна надавати можливість редагування/видалення карточки нормативу координат кривих водосховища;

7. система повинна надавати можливість ведення норм заборів води на водогосподарських системах;

7.1. система повинна надавати можливість створення карточки норм заборів води на водогосподарській системі;

7.1.1. система повинна надавати можливість вибору каналу або зрошувальної системи, внесення декадного плану та сезонів робіт;

7.2. система повинна надавати можливість редагування карточки норм заборів води на водогосподарській системі;

7.3. система повинна надавати можливість видалення карточки норм заборів води на водогосподарській системі;

8. система повинна надавати можливість ведення нормативів рівнів та витрат води на контрольних пунктах;

8.1. система повинна надавати можливість створення карточки нормативів рівнів та витрат води на контрольному пункті;

8.1.1. система повинна надавати можливість вибору річки-пункту, внесення «0» поста та норм витрат для кожного місяця;

8.2. система повинна надавати можливість редагування карточки нормативів рівнів та витрат води на контрольному пункті;

8.3. система повинна надавати можливість видалення карточки нормативів рівнів та витрат води на контрольному пункті;

9. система повинна надавати можливість ведення нормативів гідрохімічних показників;

9.1. система повинна надавати можливість створення карточки нормативу гідрохімічного показника;

9.1.1. система повинна надавати можливість вибору показника та внесення норми;

9.2. система повинна надавати можливість редагування карточки нормативу гідрохімічного показника;

9.3. система повинна надавати можливість видалення карточки нормативу гідрохімічного показника;

10. система повинна надавати можливість ведення контролю водогосподарської обстановки;

10.1. система повинна надавати можливість створення карточки контролю водогосподарської обстановки за конкретну дату;

10.1.1. система повинна надавати можливість введення фактичних показників гідрологічної ситуації на водосховищах із автоматичним заповненням нормативних та розрахованих параметрів;

10.1.2. система повинна надавати можливість завантаження фактичних показників гідрологічної ситуації на водосховищах з файлу визначеного формату;

10.1.3. система повинна надавати можливість введення фактичних показників стану водогосподарських систем з автоматичним заповненням нормативних та розрахованих показників;

10.1.4. система повинна надавати можливість введення фактичних показників гідрологічної ситуації на контрольних пунктах з автоматичним заповненням нормативних значень;

10.1.5. система повинна надавати можливість введення фактичних показників гідрохімічної обстановки річки-пункту із автоматичним заповненням нормативних значень;

10.2. система повинна надавати можливість редагування карточки контролю водогосподарської обстановки за конкретну дату;

10.3. система повинна надавати можливість видалення карточки контролю водогосподарської обстановки за конкретну дату;

					ДП ІС-5109.1181-с.ТЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

11. система повинна надавати можливість ведення довідників;

11.1. система повинна надавати можливість додавати значення в довідник;

11.2. система повинна надавати можливість редагування/видалення значення довідника;

12. система повинна надавати можливість формування звітності по даним моніторингу;

12.1. система повинна надавати можливість задання параметрів формування звітності;

13. система повинна надавати можливість реєстрації користувачів та визначення їхніх ролей.

13.1. система повинна надавати можливість видалення/редагування даних користувачів;

14. система повинна надавати можливість входу в систему.

#### 4.2 Вимоги до надійності

Інформаційна система має наступні вимоги до надійності:

- інформаційна система повинна використовувати криптографічні бібліотеки для застосування механізмів ідентифікації та аутентифікації;
- в разі виникнення позаштатних ситуації, аварій, відмов технічного забезпечення чи збоїв у зальносистемному програмному забезпеченні, система повинна мати можливість відновлювати свою працездатність з резервних копій за короткий проміжок часу та з мінімальною втратою інформації.

#### 4.3 Вимоги до складу і параметрів технічних засобів

Для забезпечення нормального функціонування системи необхідно забезпечити наступний набір технічних засобів:

					ДП ІС-5109.1181-с.ТЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		



- мінімальна конфігурація сервера:
  - 1) 2-ядерний процесор з тактовою частотою від 1.4 ГГц;
  - 2) об'єм оперативної пам'яті не менше 3 Гб;
  - 3) наявність вільного дискового простору не менше 15 Гб;
  - 4) серверний антивірус;
- мінімальна конфігурація комп'ютера-клієнта:
  - 1) процесор з тактовою частотою від 1.4 ГГц;
  - 2) об'єм оперативної пам'яті не менше 2 Гб;
  - 3) наявність вільного дискового простору не менше 5 Гб;
  - 4) клієнтське програмне забезпечення на базі Microsoft Office 2007 і вище;
  - 5) браузері клієнтських робочих станцій - Microsoft Internet Explorer версії 9.0 або вище, Google Chrome, Mozilla Firefox;
  - 6) антивірус для робочих станцій.

## 5 СТАДІЇ І ЕТАПИ РОЗРОБКИ

Основні етапи виконання робіт з розробки інформаційної системи моніторингу та екологічної оцінки водних ресурсів України.

№ з/п	Назва етапів виконання дипломного проекту	Строк виконання
1.	Вивчення рекомендованої літератури	17.04.2019
2.	Аналіз існуючих методів розв'язання задачі	18.04.2019
3.	Постановка та формалізація задачі	19.04.2019
4.	Розробка інформаційного забезпечення	21.04.2019
5.	Алгоритмізація задачі	25.04.2019
6.	Обґрунтування використовуваних технічних засобів	26.04.2019
7.	Розробка програмного забезпечення	18.05.2019
8.	Налагодження програми	19.05.2019

## 6 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ СИСТЕМИ

### 6.1 Види випробувань

Прийом робіт виконується шляхом проведення приймальних випробувань. Випробування проводяться у відповідності до розробленої Виконавцем та затвердженої Замовником Програми та методики проведення випробувань.

Перед початком проведення приймальних випробувань комісії надаються такі документи:

- технічне завдання;
- керівництво користувача;
- експлуатаційна документація;
- програма та методика випробувань.

Приймальні випробування виконуються приймальною комісією у складі уповноважених осіб Замовника та Виконавця. По результатах оформлюється Акт здачі-приймання робіт. Мета приймальних випробувань полягає у підтвердженні відповідності проведених робіт затвердженому Технічному завданню. Приймальні випробування здійснюються на базі технічних засобів Замовника.

					ДП ІС-5109.1181-с.ТЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16



НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”  
Кафедра автоматизованих систем обробки інформації та управління

**УЗГОДЖЕНО**

**Керівник проекту**

\_\_\_\_\_  
(підпис) М.О. Сперкач  
(ініціали, прізвище)

“13” травня 2019 р.

**ЗАТВЕРДЖУЮ**

**В.о. завідувача кафедри**

\_\_\_\_\_  
(підпис) О.А.Павлов  
(ініціали, прізвище)

“14” травня 2019 р.

Інформаційна система моніторингу та екологічної оцінки водних  
ресурсів України

**ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАНЬ**

Шифр ДП ІС-5109.1181-с.ПМВ

на 25 сторінках

Київ – 2019 року

## ЗМІСТ

1	ОБ'ЄКТ ВИПРОБУВАННЯ .....	3
1.1	Найменування програми .....	3
1.2	Область застосування.....	3
1.3	Умовне позначення програми .....	3
2	МЕТА ВИПРОБОВУВАНЬ .....	4
3	Вимоги до програмного продукту.....	5
3.1	Вимоги до функціональних характеристик .....	5
3.1.1	Вимоги до складу виконуваних функцій .....	5
4	Вимоги до програмної документації.....	11
5	Склад і порядок випробувань .....	12
6	Методи випробувань .....	13

					ДП ІС-5109.1393-с.ПМВ							
Зм.	Арк.	Прізвище	Підпис	Дата								
Розроб.		Казмірчук А.В			Інформаційна система моніторингу та екологічної оцінки водних ресурсів України			Літ.		Лист	Листів	
										2	25	
Перевірив.		Сперкач М.О.						КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІС-51				
Н. кон.		Телишева Т.О.										
Затв.		Павлов О.А.										

## 1 ОБ'ЄКТ ВИПРОБУВАННЯ

### 1.1 Найменування програми

Повне найменування системи «Інформаційна система моніторингу та екологічної оцінки водних ресурсів України».

### 1.2 Область застосування

Програма розроблена для Держводагентства для підтримки процесів ведення моніторингу постів спостереження та контролю водогосподарської обстановки, обробки отриманих даних для підвищення оперативності та ефективності аналізу стану водних ресурсів.

Результатом цього буде покращення інформаційного супроводу моніторингу поверхневих вод України, зменшення часу аналізу та формування звітності для подальшого прийняття рішень.

### 1.3 Умовне позначення програми

Умовне позначення програми «Інформаційна система моніторингу та екологічної оцінки водних ресурсів України».

## 2 МЕТА ВИПРОБУВАНЬ

Мета приймальних випробувань полягає у підтвердженні відповідності проведених робіт затвердженому Технічному завданню.

					ДП ІС-5109.1181-с.ПМВ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		



### 3 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

#### 3.1 Вимоги до функціональних характеристик

Система повинна містити наступні функції:

- реєстрація пунктів спостереження;
- ведення моніторингу пунктів спостереження;
- ведення контролю водогосподарської обстановки;
- створення карти перевищення ГДК показників на постах спостереження;
- створення карти розміщення пунктів спостереження;
- заповнення довідників;
- заповнення ГДК;
- внесення нормативів;
- формування звітності;
- керування користувачами системи.

##### 3.1.1 Вимоги до складу виконуваних функцій

Система повинна виконувати наступні функції:

1. система повинна надавати можливість реєстрації поста спостереження;

1.1. система повинна надавати можливість створення карточки поста спостереження;

1.1.1. система повинна надавати можливість введення всіх необхідних атрибутів поста спостереження;

1.1.2. система повинна надавати можливість визначення координат поста спостереження для відображення на карті;

1.1.3. система повинна надавати можливість вибору показників для моніторингу певного поста спостереження;

- 1.2. система повинна надавати можливість редагування карточки поста спостереження;
- 1.3. система повинна надавати можливість видалення карточки поста спостереження;
2. система повинна надавати можливість ведення моніторингу поста спостереження;
  - 2.1. система повинна надавати можливість створення карточки моніторингу поста спостереження;
    - 2.1.1. система повинна надавати можливість вибору поста спостереження;
    - 2.1.2. система повинна надавати можливість внесення даних до таблиці спостережень за конкретну дату спостереження;
    - 2.1.3. система повинна надавати можливість редагування даних таблиці спостережень за конкретну дату спостережень;
    - 2.1.4. система повинна надавати можливість видалення даних з таблиці спостережень за конкретну дату спостережень;
  - 2.2. система повинна надавати можливість редагування карточки моніторингу поста спостереження;
  - 2.3. система повинна надавати можливість видалення карточки моніторингу поста спостереження;
3. система повинна надавати можливість ведення нормативів ГДК;
  - 3.1. система повинна надавати можливість створення карточки нормативів ГДК для показника;
    - 3.1.1. система повинна надавати можливість вибору групи показників;
    - 3.1.2. система повинна надавати можливість вибору показника;
    - 3.1.3. система повинна надавати можливість додавання нормативів показника з мінімальним та максимальним допустимим значенням;

- 3.1.4. система повинна надавати можливість редагування/видалення нормативів показника;
- 3.2. система повинна надавати можливість редагування карточки нормативу ГДК;
- 3.3. система повинна надавати можливість видалення карточки нормативу ГДК;
4. система повинна надавати можливість перегляду карти перевищення ГДК показників на постах спостереження;
- 4.1. система повинна відображати зареєстровані пости спостереження на карті;
- 4.1.1. система повинна відображати пости спостереження маркерами, колір і розмір яких відповідає рівню і кількості перевищень ГДК показників;
- 4.1.2. система повинна при наведенні на маркер надавати коротку реєстраційну інформацію про пост спостереження і значення показників на останню дату спостережень;
- 4.1.3. система повинна при виборі поста спостереження на карті надавати можливість переходу на формування звіту по даних моніторингу даного поста спостереження;
5. система повинна надавати можливість перегляду карти розміщення постів спостереження;
- 5.1. система повинна відображати зареєстровані пости спостереження на карті;
- 5.1.1. система повинна при наведенні на відповідний пост спостереження, що розміщений на карті, надавати коротку реєстраційну інформацію про пост спостереження;
- 5.1.2. система повинна при виборі поста спостереження на карті надавати можливість переходу на формування звіту по даних моніторингу даного поста спостереження;

6. система повинна надавати можливість ведення нормативів для контролю водогосподарської обстановки;

6.1. система повинна надавати можливість створення карточки нормативу координат кривих водосховища;

6.1.1. система повинна надавати можливість вибору основного басейна, водосховища, внесення проектного рівня та проектного обсягу, координат кривих об'ємів водосховища;

6.2. система повинна надавати можливість редагування/видалення карточки нормативу координат кривих водосховища;

7. система повинна надавати можливість ведення норм заборів води на водогосподарських системах;

7.1. система повинна надавати можливість створення карточки норм заборів води на водогосподарській системі;

7.1.1. система повинна надавати можливість вибору каналу або зрошувальної системи, внесення декадного плану та сезонів робіт;

7.2. система повинна надавати можливість редагування карточки норм заборів води на водогосподарській системі;

7.3. система повинна надавати можливість видалення карточки норм заборів води на водогосподарській системі;

8. система повинна надавати можливість ведення нормативів рівнів та витрат води на контрольних пунктах;

8.1. система повинна надавати можливість створення карточки нормативів рівнів та витрат води на контрольному пункті;

8.1.1. система повинна надавати можливість вибору річки-пункту, внесення «0» поста та норм витрат для кожного місяця;

8.2. система повинна надавати можливість редагування карточки нормативів рівнів та витрат води на контрольному пункті;

8.3. система повинна надавати можливість видалення карточки нормативів рівнів та витрат води на контрольному пункті;

9. система повинна надавати можливість ведення нормативів гідрохімічних показників;

9.1. система повинна надавати можливість створення карточки нормативу гідрохімічного показника;

9.1.1. система повинна надавати можливість вибору показника та внесення норми;

9.2. система повинна надавати можливість редагування карточки нормативу гідрохімічного показника;

9.3. система повинна надавати можливість видалення карточки нормативу гідрохімічного показника;

10. система повинна надавати можливість ведення контролю водогосподарської обстановки;

10.1. система повинна надавати можливість створення карточки контролю водогосподарської обстановки за конкретну дату;

10.1.1. система повинна надавати можливість введення фактичних показників гідрологічної ситуації на водосховищах із автоматичним заповненням нормативних та розрахованих параметрів;

10.1.2. система повинна надавати можливість завантаження фактичних показників гідрологічної ситуації на водосховищах з файлу визначеного формату;

10.1.3. система повинна надавати можливість введення фактичних показників стану водогосподарських систем з автоматичним заповненням нормативних та розрахованих показників;

10.1.4. система повинна надавати можливість введення фактичних показників гідрологічної ситуації на контрольних пунктах з автоматичним заповненням нормативних значень;

10.1.5. система повинна надавати можливість введення фактичних показників гідрохімічної обстановки річки-пункту із автоматичним заповненням нормативних значень;

10.2. система повинна надавати можливість редагування карточки контролю водогосподарської обстановки за конкретну дату;

10.3. система повинна надавати можливість видалення карточки контролю водогосподарської обстановки за конкретну дату;

11. система повинна надавати можливість ведення довідників;

11.1. система повинна надавати можливість додавати значення в довідник;

11.2. система повинна надавати можливість редагування/видалення значення довідника;

12. система повинна надавати можливість формування звітності по даним моніторингу;

12.1. система повинна надавати можливість задання параметрів формування звітності;

13. система повинна надавати можливість реєстрації користувачів та визначення їхніх ролей.

13.1. система повинна надавати можливість видалення/редагування даних користувачів;

14. система повинна надавати можливість входу в систему.

#### 4 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

Програмна документація має бути розроблена згідно нормативних документів, що наведені в технічному завданні і містити наступні документи:

- технічне завдання;
- пояснювальна записка;
- графічний матеріал;
- програма та методика випробувань.

Випробовування проводяться на основі нижченаведених документів:

- ГОСТ 34.603-92. Інформаційна технологія. Види випробувань автоматизованих систем;
- ГОСТ РД 50-34.698-90. Автоматизовані системи вимог до змісту документів.

					ДП ІС-5109.1181-с.ПМВ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

**5 СКЛАД І ПОРЯДОК ВИПРОБУВАНЬ**

Приймальні випробування виконуються приймальною комісією у складі уповноважених осіб Замовника та Виконавця. Випробовування проводяться методом ручного тестування. По результатам оформлюється Акт здачі-приймання робіт. Приймальні випробування здійснюються на базі технічних засобів Замовника.

					ДП ІС-5109.1181-с.ПМВ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		



## 6 МЕТОДИ ВИПРОБУВАНЬ

Під час тестування була перевірена основна функціональність комплексу задач (КЗ). В таблицях 6.1-6.11 наведено перелік випробувань основних функціональних можливостей.

Таблиця 6.1 – Вхід в систему

Мета тесту	Перевірка функції «Вхід в систему»
Початковий стан КЗ	Відкрита сторінка для входу в систему
Вхідні данні	Логін та пароль користувача
Схема проведення тесту	Ввести у поле «Логін» логін користувача, а у поле «Пароль» - пароль користувача. Натиснути кнопку «Вхід»
Очікуваний результат	Відкрита головна сторінка із пунктами меню із врахуванням прав доступу поточного користувача
Стан КЗ після проведення випробувань:	Відкрита головна сторінка із пунктами меню із врахуванням прав доступу поточного користувача

Таблиця 6.2 – Перевірка заповнення логіна і пароля

Мета тесту	Перевірка функції «Перевірка заповнення логіна та пароля»
Початковий стан КЗ	Відкрита сторінка для входу в систему
Вхідні дані	Логін або пароль
Схема проведення тесту	Ввести у поле «Логін» логін користувача, а поле «Пароль» - залишити пустим. Або навпаки. Натиснути кнопку «Вхід»

Продовження таблиці 6.2

Мета тесту	Перевірка функції «Перевірка заповнення логіна та пароля»
Очікуваний результат	Кнопка «Вхід» недоступна для натискання. Біля пустих полів повідомлення валідації «Поле обов'язкове для заповнення». Вхід в систему не виконано
Стан КЗ після проведення випробувань	Відкрита сторінка для входу в систему

Таблиця 6.3 – Перевірка правильності введення логіна і пароля

Мета тесту	Перевірка функції «Перевірка правильності введення логіна і пароля»
Початковий стан КЗ	Відкрита сторінка для входу в систему
Вхідні данні	Логін та пароль
Схема проведення тесту	Ввести у поле «Логін» неправильний логін, у поле «Пароль» ввести неправильний пароль, натиснути кнопку «Вхід»
Очікуваний результат	Відкрита сторінка для входу в систему із повідомленням «Невірно було вказані логін або пароль». Вхід в систему не виконано
Стан КЗ після проведення випробувань	Відкрита сторінка для входу в систему

Таблиця 6.4 – Перевірка внесення даних моніторингу

Мета тесту	Перевірка функції «Внесення даних моніторингу»
Початковий стан КЗ	Відкрита сторінка моніторингу поста спостереження
Вхідні дані	Виміряні значення показників
Схема проведення тесту	Натиснути кнопку «Додавання нового запису». Заповнити поле «Дата внесення проби», заповнити інші поля виміряними значеннями показників. Натиснути кнопку збереження
Очікуваний результат	Новий рядок із введеними даними успішно збережено
Стан КЗ після проведення випробування	Відкрита сторінка моніторингу поста спостереження

Таблиця 6.5 – Валідація введених значень показників

Мета тесту	Перевірка функції «Валідація введених значень показників»
Початковий стан КЗ	Відкрита сторінка моніторингу поста спостереження
Вхідні дані	Виміряні значення показників
Схема проведення тесту	Натиснути кнопку «Додавання нового запису». Заповнити поле «Дата внесення проби», в поле будь-якого показника записати символ або число в неправильному форматі. Натиснути кнопку збереження рядка

Продовження таблиці 6.5

Мета тесту	Перевірка функції «Валідація введених значень показників»
Очікуваний результат	Повідомлення про помилку валідації «Значення показника має бути у форматі (12,5)». Рядок не збережено
Стан КЗ після проведення випробування	Відкрита сторінка моніторингу поста спостереження

Таблиця 6.6 – Перевірка редагування доданого запису в таблиці спостережень

Мета тесту	Перевірка функції «Перевірка редагування доданого запису в таблиці спостережень»
Початковий стан КЗ	Відкрита сторінка моніторингу поста спостереження
Вхідні дані	Виміряні значення показників
Схема проведення тесту	Натиснути кнопку «Редагування запису». Відредагувати будь-яке значення поля показника. Натиснути кнопку збереження рядка
Очікуваний результат	Рядок успішно відредаговано
Стан КЗ після проведення випробування	Відкрита сторінка моніторингу поста спостереження

Таблиця 6.7 – Видалення запису з таблиці спостережень

Мета тесту	Перевірка функції «Видалення запису з таблиці спостережень»
Початковий стан КЗ	Відкрита сторінка моніторингу поста спостереження
Вхідні дані	Немає
Схема проведення тесту	Натиснути кнопку «Видалення запису»
Очікуваний результат	Рядок успішно видалено
Стан КЗ після проведення випробування	Відкрита сторінка моніторингу поста спостереження

Таблиця 6.8 – Перегляд короткої інформації про пост спостереження на карті «Перевищення ГДК показників якості води на постах спостереження»

Мета тесту	Перевірка функції «Перегляд інформації про пост спостереження на карті»
Початковий стан КЗ	Відкрита сторінка з картою «Перевищення ГДК показників якості води на постах спостереження»
Вхідні дані	Немає
Схема проведення тесту	Натиснути або навести курсор на будь-який індикатор, що відображає стан поста спостереження.

## Продовження таблиці 6.8

Очікуваний результат	З'являється спливаюче віконечко з короткою інформацією про даний пост спостереження та табличкою із значеннями показників у порівнянні з ГДК на останню дату спостереження.
Стан КЗ після проведення випробування	Відкрита сторінка з картою «Перевищення ГДК показників якості води на постах спостереження»

Таблиця 6.9 – Перевірка посилання на формування звіту по даним моніторингу

<b>Мета тесту</b>	<b>Перевірка функції «Перевірка посилання на формування звіту по даним моніторингу»</b>
Початковий стан КЗ	Відкрита сторінка з картою «Перевищення ГДК показників якості води на постах спостереження»
Вхідні дані	Немає
Схема проведення тесту	Натиснути на будь-який індикатор, що відображає стан поста спостереження. З'являється спливаюче віконечко з короткою інформацією про даний пост спостереження, табличкою із значеннями показників у порівнянні з ГДК на останню дату спостереження та посилання на формування звіту із назвою «Дані моніторингу». Переходимо за даним посиланням

Мета тесту	Перевірка функції «Перевірка посилання на формування звіту по даним моніторингу»
Очікуваний результат	Відкривається сторінка для введення параметрів формування звітності – «Період з», «Період до», «Показник»
Стан КЗ після проведення випробування	Відкривається сторінка для введення параметрів формування звітності – «Період з», «Період до», «Показник»

Продовження таблиці 6.9

Таблиця 6.10 – Перевірка формування звіту по даним моніторингу на пості спостереження

Мета тесту	Перевірка функції «Формування звіту по даним моніторингу на пості спостереження»
Початковий стан КЗ	Сторінка для введення параметрів формування звітності – «Період з», «Період до», «Показник»
Вхідні дані	Період з, період до, показник
Схема проведення тесту	Заповнюємо поля «Період з», «Період до», «Показник». Натискаємо кнопку «Виконати».
Очікуваний результат	Сформовано звіт у таблично-графічному вигляді по вибраним показникам за вказаний період моніторингу

Продовження таблиці 6.10

Мета тесту	Перевірка функції «Формування звіту по даним моніторингу на пості спостереження»
Стан КЗ після проведення випробування	Сформовано звіт у таблично-графічному вигляді по вибраним показникам за вказаний період моніторингу

Таблиця 6.11 – Перевірка реєстрації пункту спостереження

Мета тесту	Перевірка функції «Реєстрація пункту спостереження»
Початковий стан КЗ	Відкрита сторінка внесення даних для реєстрації нового пункту спостереження
Вхідні дані	Заповнені всі обов'язкові поля
Схема проведення тесту	Заповнити всі обов'язкові поля (позначені зірочкою) і натиснути кнопку «Зберегти»
Очікуваний результат	Дані успішно зберігаються, сторінка перезавантажується і одразу відкривається в режимі редагування.
Стан КЗ після проведення випробування	Відкрита сторінка на редагування щойно доданого пункту спостереження



Таблиця 6.12 – Валідація даних при реєстрації пункту спостереження

Мета тесту	Перевірка функції «Валідація даних при реєстрації пункту спостереження»
Початковий стан КЗ	Відкрита сторінка внесення даних для реєстрації нового пункту спостереження
Вхідні дані	Лінійна координата (відстань до гирла), км
Схема проведення тесту	Введення в поле букв
Очікуваний результат	Поле «Лінійна координата (відстань до гирла), км» підсвічується і під ним розташований текст повідомлення про помилку червоного кольору «Значення поля 'Лінійна координата (відстань до гирла), км' не є дійсним числом формату {5,1}»
Стан КЗ після проведення випробування	Відкрита сторінка внесення даних для реєстрації нового пункту спостереження, поле «Лінійна координата (відстань до гирла), км» підсвічується і під ним розташований текст повідомлення про помилку червоного кольору «Значення поля 'Лінійна координата (відстань до гирла), км' не є дійсним числом формату {5,1}»

Таблиця 6.13 – Валідація даних при реєстрації нового користувача

<b>Мета тесту</b>	<b>Перевірка функції «Валідація даних при реєстрації нового користувача»</b>
Початковий стан КЗ	Відкрита сторінка реєстрації нового користувача
Вхідні дані	Пароль, Повторіть пароль

Продовження таблиці 6.13

<b>Мета тесту</b>	<b>Перевірка функції «Валідація даних при реєстрації нового користувача»</b>
Схема проведення тесту	В поле «Пароль» вводимо значення «123456», в поле «Повторіть пароль» вносимо значення «654321»
Очікуваний результат	Під полем «Пароль» розташоване повідомлення «Пароль і підтвердження пароля не збігаються».
Стан КЗ після проведення випробування	Відкрита сторінка реєстрації нового користувача і під полем «Пароль» розташоване повідомлення «Пароль і підтвердження пароля не збігаються»

Таблиця 6.14 – Перевірка валідації даних при внесенні ГДК

<b>Мета тесту</b>	<b>Перевірка функції «Валідація даних при внесенні ГДК»</b>
Початковий стан КЗ	Відкрита сторінка «Нормативи ГДК» для внесення нормативу показника
Вхідні дані	Показник
Схема проведення тесту	В довідникове поле «Показник» введемо будь-яке значення і натискаємо кнопку «Зберегти»

<b>Мета тесту</b>	<b>Перевірка функції «Валідація даних при внесенні ГДК»</b>
Очікуваний результат	Карточка не зберігається, під полем «Показник» розташований текст повідомлення про помилку червоного кольору «Значення поля 'Показник' необхідно обрати із списку»

Продовження таблиці 6.14

<b>Мета тесту</b>	<b>Перевірка функції «Валідація даних при внесенні ГДК»</b>
Стан КЗ після проведення випробування	Відкрита сторінка «Нормативи ГДК» для внесення нормативу показника і під полем «Показник» розташований текст повідомлення про помилку червоного кольору «Значення поля 'Показник' необхідно обрати із списку»

Таблиця 6.15 – Перевірка завантаження даних по гідрологічній ситуації на водосховищах з файлу

<b>Мета тесту</b>	<b>Перевірка функції «Завантаження даних по гідрологічній ситуації на водосховищах з файлу»</b>
Початковий стан КЗ	Сторінка «Гідрологічна ситуація на водосховищах» для додавання даних на конкретну дату спостереження

<b>Мета тесту</b>	<b>Перевірка функції «Завантаження даних по гідрологічній ситуації на водосховищах з файлу»</b>
Вхідні дані	Файл з розширенням .doc у правильному форматі
Схема проведення тесту	Натискаємо кнопку «Додати файл». Вибираємо файл правильного формату і завантажуюмо його. Після завантаження файлу натискаємо кнопку «Завантаження даних»

Продовження таблиці 6.15

<b>Мета тесту</b>	<b>Перевірка функції «Завантаження даних по гідрологічній ситуації на водосховищах з файлу»</b>
Очікуваний результат	У таблицю «Гідрологічна ситуація на водосховищі» додано записи по кожному водосховищу із всіма параметрами і розраховано Вільний обсяг
Стан КЗ після проведення випробування	Сторінка «Гідрологічна ситуація на водосховищах» для додавання даних на конкретну дату спостереження

Таблиця 6.16 – Перевірка завантаження даних по гідрологічній ситуації на водосховищах із документу неправильного формату.

Мета тесту	Перевірка функції «Обробка випадку завантаження даних по гідрологічній ситуації на водосховищах з файлу неправильного формату»
Початковий стан КЗ	Сторінка «Гідрологічна ситуація на водосховищах» для додавання даних на конкретну дату спостереження
Вхідні дані	Довільний файл з розширенням .doc , що не відповідає заданому формату
Схема проведення тесту	Натискаємо кнопку «Додати файл». Вибираємо файл і завантажуюємо його. Після завантаження файлу натискаємо кнопку «Завантаження даних»

Продовження таблиці 6.16

Мета тесту	Перевірка функції «Обробка випадку завантаження даних по гідрологічній ситуації на водосховищах з файлу неправильного формату»
Очікуваний результат	Повідомлення «Файл, що оброблюється, не відповідає заданій структурі!». У таблицю «Гідрологічна ситуація на водосховищі» не додано нових записів
Стан КЗ після проведення випробування:	Сторінка «Гідрологічна ситуація на водосховищах» для додавання даних на конкретну дату спостереження

# **Графічний матеріал до дипломного проекту**

на тему: Інформаційна система моніторингу та екологічної оцінки  
водних ресурсів України

---

Київ – 2019 року